



E-Covid-19 Support

Submitted by

Kamrun Nahar Liza

ID: 172-35-2144

Department of Software Engineering

Daffodil International University

Supervised by

Asif Khan Shakir

Lecturer (Senior Scale)

Faculty of Science and Information Technology

Department of Software Engineering

Daffodil International University

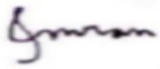
Fall- 2020

A project (SWE 431) submitted in fulfillment of the requirements for the degree
of BSc. in Software Engineering

APPROVAL

This project named “E-covid-19 Support” submitted by **Kamrun Nahar Liza, ID: 172-35-2144** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillments of the requirements of the degree of Bachelor of Science in Software Engineering and approved as its style and contents.

BOARD OF EXAMINERS



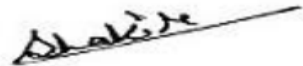
Dr. Imran Mahmud
Associate Professor and Head
Department of Software Engineering
Daffodil International University

Chairman



Md Anwar Hossen
Assistant Professor
Department of Software Engineering
Daffodil International University

Internal Examiner 1



Asif Khan Shakir
Senior Lecturer
Department of Software Engineering
Daffodil International University

Internal Examiner 2



Professor Dr M Shamim Kaiser
Institute of Information Technology
Jahangirnagar University

External Examiner

Fall-2020
DAFFODIL INTERNATIONAL UNIVERSITY

Declaration of Project:

Author Full Name : Kamrun Nahar Liza
Date of Birth : 30-10-1997
Title of Project : E-Covid-19 Support
Academic Session : 2020

I, the author, hereby declare that the project becomes the property of Daffodil International University (DIU) and to be placed at the department of Software Engineering. I also give permission to the DIU to reproduce this project in whole or in part for the purpose of research only.

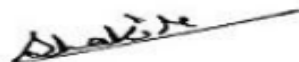


SIGNATURE

Kamrun Nahar Liza(172-35-2144)
STUDENT NAME & STUDENT ID

Date:31-03-2021

Certified by:



SIGNATURE OF SUPERVISOR

Asif Khan Shakir
NAME OF SUPERVISOR

Date: 31-03-2021

ACKNOWLEDGEMENT

First of all I am so much grateful to almighty Allah who gives me sound mind, sound health, concept and ability to complete this project successfully. I am so thankful to Daffodil International University for giving me such an opportunity to prove myself. I am also thankful to Dr. Imran Mahmud, Department Head Software Engineering for giving academic support. And lastly, I want to thank my supervisor Asif Khan Shakir for his valuable guidelines in order to work on this project.

PROJECT SUMMARY

“E-Covid-19 Support” is a web-based application where doctors can post Covid-19 symptoms and it’s treatment according to the symptoms. Patients can see the doctor's profile. Patients can also see in instruction posts of every country. Patients can see these symptoms and recovery instruction. Patients can also search details of recovery instructions by only typing his symptoms. Patients can communicate with the doctor through his recovery instruction post by commenting. Patients can also give likes or dislikes on recovery instruction posts. Patients can upload posts for plasma. In plasma posts if anyone can donate he can let him through commenting. There is also another user who is Plasma Donor. Plasma donors will make his profile. If any patient needs plasma urgently he can check the list of plasma donors and can send requests to specific plasma donors. Plasma donor can accept the request if he wants to donate, after accepting the request plasma donor’s phone number and more details will be available for the requested patient. Admin will give updates of the daily affected people of covid-19 all over the world.

Table of Contents

APPROVAL.....	i
PROJECT DECLARATION	i i
ACKNOWLEDGEMENT	iii
PROJECT SUMMARY.....	iv
TABLE OF CONTENTS	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
LIST OF ABBREVIATIONS	x
CHAPTER 1: INTRODUCTION.....	01
1.1 Project Overview.....	01
1.2 Project Purpose.....	01
1.2.1 Background	01
1.2.2 Benefits & Beneficiaries	01
1.2.3 Goals	02
1.3 Stakeholders.....	02
1.4 Data Flow Diagrams	03
1.5 Project Schedule.....	04
1.5.1 Gantt Chart	04
CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION	
2.1 Functional Requirements.....	05
2.2 Non-Functional Requirements	07
2.3 Performance Requirements.....	07
2.3.1 Speed and Latency Requirements	
2.3.2 Precision or Accuracy Requirements	
2.3.3 Capacity Requirements	
2.4 Dependability Requirements	08
2.4.1 Reliability Requirements.....	08
2.4.2 Availability Requirements.....	08

2.5 Maintainability and Supportability Requirements.....	08
2.5.1 Maintenance Requirements.....	08
2.5.2 Supportability Requirements	
2.5.3 Adaptability Requirements	
2.5.4 Scalability or Extensibility Requirements	
2.6 Security Requirements.....	09
2.6.1 Access Requirements	
2.6.2 Integrity Requirements	
2.6.3 Privacy Requirements	
2.7 Usability and Human-Interaction Requirements.....	09
2.7.1 Ease of Use Requirements	
2.7.2 Personalization and Internationalization Requirements	
2.7.3 Understandability and Politeness Requirements	
2.7.4 Accessibility Requirements	
2.7.5 User Documentation Requirements	
2.7.6 Training Requirements	
2.8 Look and Feel Requirements.....	10
2.8.1 Appearance Requirements	
2.8.2 Style Requirements	
2.9 Operational and Environmental Requirements.....	11
2.9.1 Expected Physical Environment	
2.9.2 Requirements for Interfacing with Adjacent Systems	
2.9.3 Project Requirements	
2.9.4 Release Requirements	
2.10. Legal Requirements.....	11
CHAPTER 3: SYSTEM ANALYSIS	
3.1 Use Case Diagram.....	12
3.2 Use Case Description (for each use case)	15
3.3 Activity Diagram (for each use case)	19
Chapter 4: System Design Specification	
4.1 Sequence Diagram (for each use case)	25
4.3 E-R Diagram.....	30
4.4 Class Diagram.....	31
4.5 Development Tools & Technology.....	32

4.5.1 User Interface Technology.....	32
4.5.2 Implementation Tools & Platforms.....	32
4.5.2.1 Visual Studio Code	
4.5.2.2 Xampp version 3.2.5	
4.5.2.3 Apache HTTP Server	
CHAPTER 5: SYSTEM TESTING	
5.1 Testing Features.....	33
5.1.1 Features to be tested	
5.1.2 Features not to be tested	
5.2 Testing Strategies.....	33
5.2.1 Test Approach	
5.2.2 Pass/Fail Criteria	
5.2.3 Suspension and Resumption	
5.2.4 Testing Schedule	
5.3 Test Cases.....	35
5.4 Testing Environment (hardware/software requirements)	40
CHAPTER 6: USER MANUAL	
6.1 User Manual	41
CHAPTER 7: PROJECT SUMMARY	
7.1 Github Link.....	51
7.2 Critical Evolution.....	51
7.3 Limitations.....	51
7.4 Obstacles & Achievements.....	51
7.5 Future Scope.....	51
CHAPTER 8: APPENDICS.....	52
REFERENCES	64

List of Tables

Point	Name of tables	Page no
Table 2.1	Functional Requirements	5
Table 2.2	Non-Functional Requirements	7
Table 5.1	Testing Schedule	34
Table 5.2	Test Case-1	35
Table 5.4	Test Case-2	36
Table 5.6	Test Case-3	37
Table 5.8	Test Case-4	38
Table 5.10	Test Case-5	39
Table 5.12	Test Case-6	40

List of Figures

Point	Name of figures	Page no:
Figure 1.5.1	Gant Chart	2
Figure 1.5.	Data Flow Diagram	3
Figure 3.1	Use Case	12
Figure 3.3	Activity Diagram	19
Figure 4.1	Sequence Diagram	25
Figure 4.2	ER Diagram	30
Figure 4.3	Class Diagram	31

List of Abbreviations

DIU = Daffodil International University

SWE = Software Engineering Department

CHAPTER-01

Introduction

1.1 Project Overview

E-Covid-19 Support is a web-based system. Through, this website Doctors will input symptoms of covid and according to those symptoms the recovery instructions from any country. Patient can search for his symptoms and get the details of symptoms and recovery instructions post. Patient can post for plasma if he needed and other patient and plasma donor can let him know by commenting if he able to donate. Patient can also send request to plasma donor for plasma. If plasma donor has accepted his request then his personal phone number will be available to that patient. Admin will update daily report of effected people of Covid-19 all over the world.

1.2 Project Purpose

1.2.1 Background

Normally when a patient is sick, he has to move to a doctor's chamber and take appointment for his checkup. These appointments must be scheduled to doctor's according time. There is no guarantee that the patient will get an appointment in his/her desired date as that date may already be booked. It is a lengthy process. But now because of this covid pandemic situation people can't get available doctor or proper treatment instruction in right time. When someone is covid positive, it will be better to take treatment from home from early stage if the health condition is not too much bad. It's also difficult to find plasma though hospital who needs urgently. To solve these kinds of problem and make our life easier I am making a website named E-Covid-19 Support (ECS) to get the primary recovery procedures of Covid-19 and confirm about covid from patient's symptoms. This E-Covid-19 Support system will be give the best treatment instruction of covid by the best doctors of all over the country and help to find nearby plasma donor urgently. This will be very helpful for this pandemic situation.

1.2.2 Benefits & Beneficiaries

- ◆ Doctors
- ◆ Patient
- ◆ Plasma Donor
- ◆ Admin

1.2.3 Goals

- ❖ Doctor will post about covid-19 recovery procedure instructions.
- ❖ The system will be developed to give best treatment instructions of covid-19 from patient's symptoms by only typing his symptoms.
- ❖ Patient can let everyone know by commenting how much beneficial the recovery instruction for him.
- ❖ Patient will post for plasma donor & able to find.

1.3 Stakeholders

- Doctors
- Patient
- Plasma Donor
- Admin

1.4 Project Schedule

1.4.1 Gant Chart

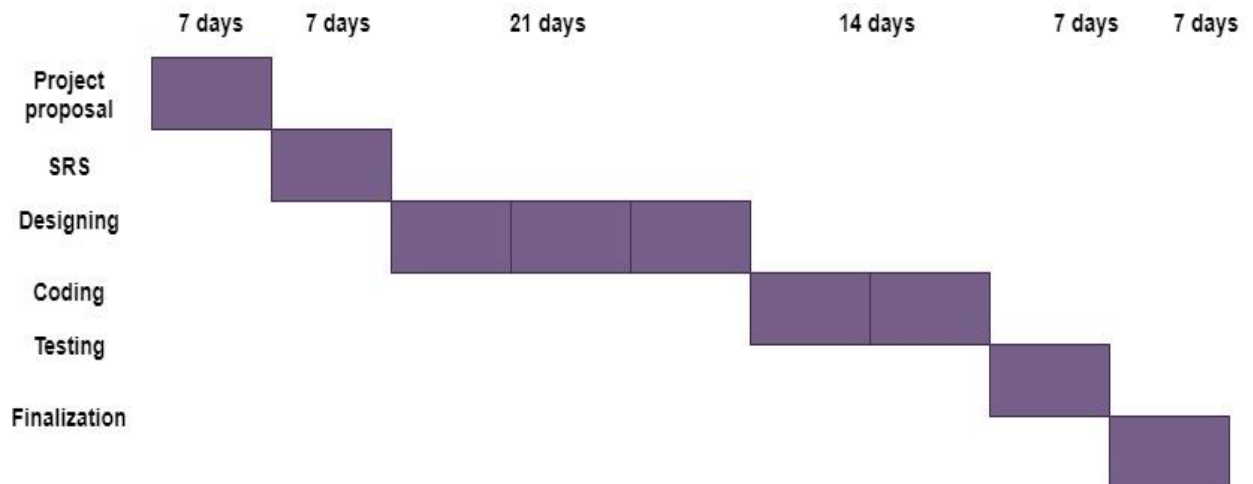


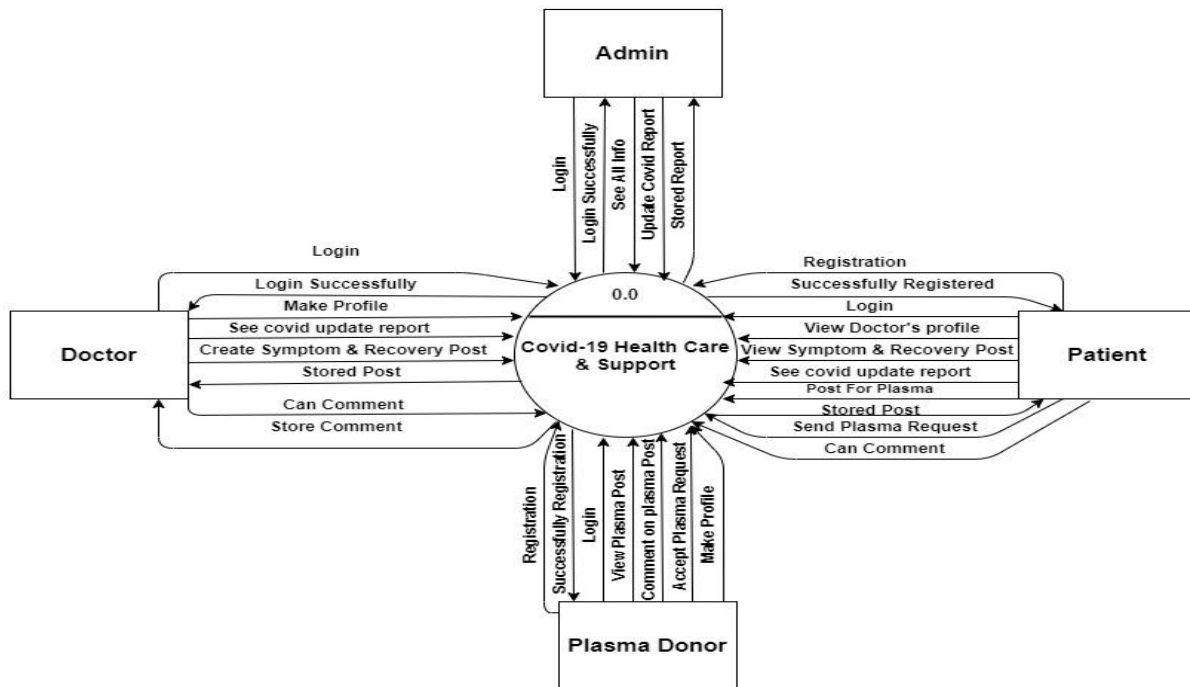
Figure 1.4.1: Gant Chart

Project Proposal	Kamrun Nahar Liza	7days
Software Requirement Specification	Kamrun Nahar Liza	7 days
Software Design	Kamrun Nahar Liza	21 days
Coding	Kamrun Nahar Liza	14 days
Software Testing	Kamrun Nahar Liza	7 days
Project Finalization	Kamrun Nahar Liza	7 days

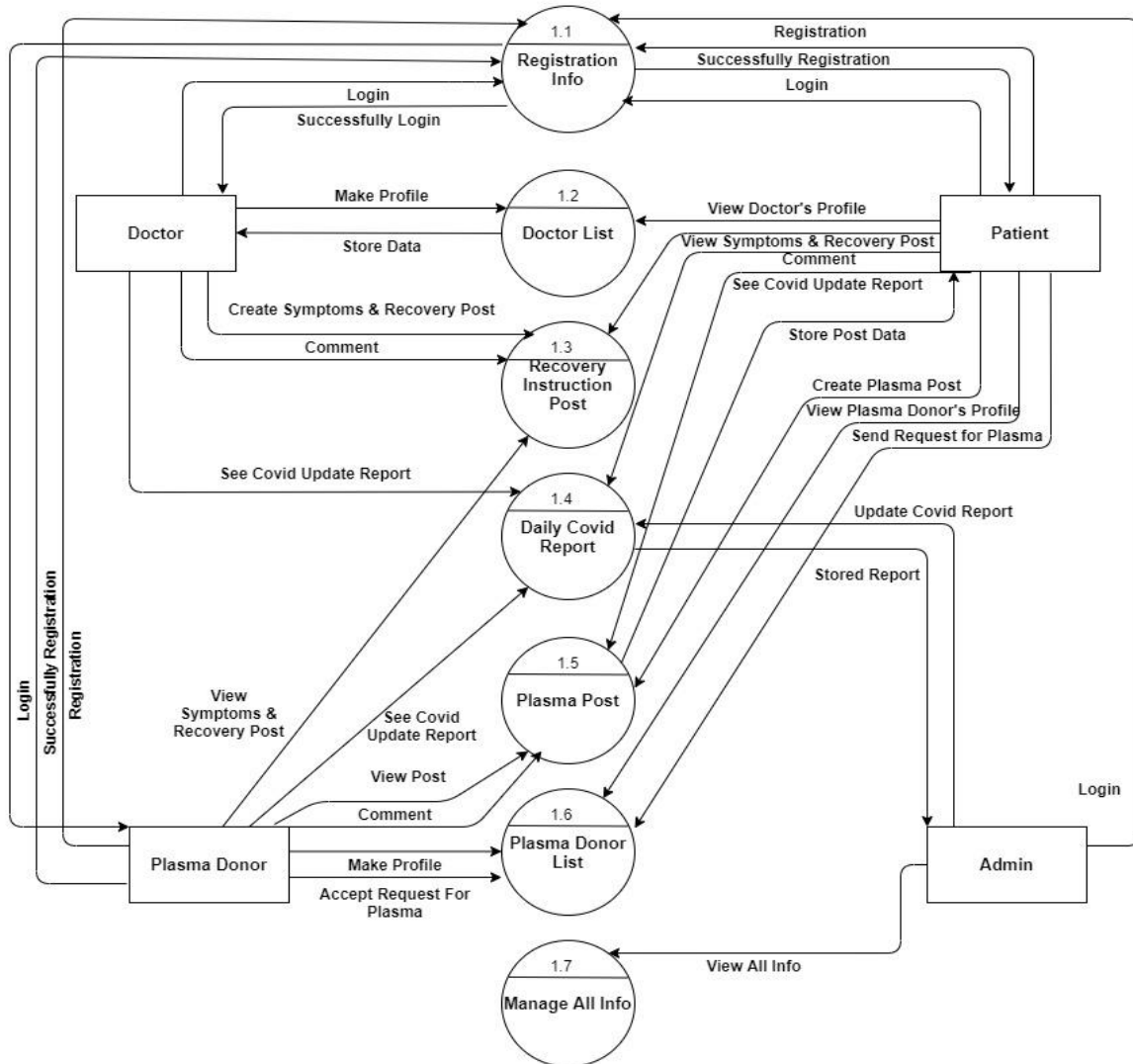
Figure 1.4.2: Work Distribution

1.5 Data Flow Diagram

1.5.1 Level 0



1.5.2 Level 1



CHAPTER-02

Software Requirement Specification

2.1 Functional Requirements

These are my project's functional requirements. Here I am explaining about my project functional requirements in details.

Table2.1: Functional Requirements

ID	Name	Description	Functional / Non functional	Priority
FRQ-01	Registration	Users have to registration in the system with his proper information.	Functional	High
FRQ-02	Login	User can login with his valid email id and password.	Functional	High
FRQ-03	Make Doctor's Profile	Doctor's can make his profile after login.	Functional	High
FRQ-04	Can see doctor's profile.	Users can see doctor's profile information.	Functional	High
FRQ-05	Create Symptoms & Recovery Post	Doctor can add Symptoms and Recovery instructions for covid-19 to create a post.	Functional	High
FRQ-06	View Symptoms & Recovery Post	All user can view symptoms and recovery post.	Functional	High
FRQ-07	Can Comment, Like or Dislike on doctor's post	Patients, Plasma Donor and Other doctors can comment, like or dislike on doctor's post.	Functional	High
FRQ-08	Search Covid-19 Instruction Post From Symptoms	Patient can search by inputting his symptoms that which Instructions post are contains these symptoms.	Functional	High

FRQ-09	Post for Plasma	Patient can make post for plasma if he need help.	Functional	High
FRQ-10	View Post for plasma	Patient and plasma donor can view these posts.	Functional	High
FRQ-11	Comment on plasma post	Patient and plasma donor can comment on plasma post.	Functional	High
FRQ-12	Make Plasma Donor's Profile	Plasma Donor can make his profile with details after login into the system.	Functional	High
FRQ-13	Send request to plasma donor	Patient can send request to plasma donor if he needs plasma urgently.	Functional	High
FRQ-14	Accept request	After getting plasma request from patient plasma donor can accept his request for viewing his contact number.	Functional	High
FRQ-15	Verify doctor	Admin will verify doctor from his doctor code and give access into the system.	Functional	High
FRQ-16	Input Covid-19 Daily Report	Admin will input the daily update of total effected number.	Functional	High
FRQ-17	View Covid-19 Daily Report	All users can see the daily report.	Functional	High

2.2 Non-Functional Requirement

Here I explain about my project non-functional requirements with details.

Table2.2: Non-Functional Requirements

ID	Name	Description	Non-Functional Requirement	Priority
NFR-01	Security	This is for use token-based authentication, session, validation 2FA and it will be more secure from unauthorized access for this system.	Non-Functional	High
NFR-02	Availability	The system will available for 24/7 for this user will get access and services whenever he wants.	Non-Functional	High
NFR-03	Accuracy	Data or process requirement concerned about defining the precision which the solution will record or produce data.	Non-Functional	High
NFR-04	Maintenance	It is a way how much easy to support, change and enhance the entire system.	Non-Functional	Medium

2.3 Performance Requirements

2.3.1 Speed and Latency Requirements

1. In this system all dataset will be inserted in MySQL Databases. This will work by using php artisan related all the commands.
2. User Interface design-build on the user fulfills table data set and show. We have to use the database of this system by using php artisan migrate command which we will run on command prompt.

2.3.2 Accuracy Requirements

1. All users of this system (doctor, patient, plasma donor and admin) will be capable to show all accurate pages where he can show detail of Symptoms & recovery post list, Doctors list, Doctor's Profile, Plasma Donor list, All request for plasma list, Plasma Post, Search Sickness Solution, Comment list, Daily Covid-19 total effected chart.
2. Doctor can update his post symptoms or medicines.
3. Admin can update daily Covid-19 total effected chart.
4. Doctor can update his profile's information.
5. Plasma donor's can update his profile information.

2.3.3 Capacity Requirements

1. At time multiple users (doctors, patients, plasma donor & system admin) can use this system.
2. Patient can send plasma request to many Plasma donors at a time.

2.4 Dependability Requirements

2.4.1 Reliability Requirements

1. Doctors, Patients, Plasma Donors and Admin should be log in to the system using his or her valid email and password.
2. Admin can easily update daily's total number of covid-19 effected people chart.
3. Doctor can add symptoms and recovery post.
4. Doctors and Patients both can comment like or dislike on doctor's recovery post.
5. Patient can post with details if he needs plasma.
6. Patient and Plasma Donors can comment on plasma post if they can help.
7. Patient can send request to plasma donor for giving plasma.
8. Plasma donor can accept his request then patient can see his phone number.

2.4.2 Availability Requirements

1. When we are using this system at first we have to run xampp or wampp server.
2. Then we have to know command for run properly and easily.
3. This web application will run on a web browser.

2.5 Maintainability and Supportability Requirements

2.5.1. Maintenance Requirements

1. Web application did not modify or change.
2. We have to maintain all the security related issues and other works.

2.5.2 Supportability Requirements

1. When this system will not work properly, then we have to check databases and other works.
2. Update all type of security patches and others system.

2.5.3 Adaptability Requirements

1. This web application is very important in this pandemic situation of covid-19.
2. This web application can help to all doctors, patients and plasma donors then they can get all services easily.

2.5.4 Scalability or Extensibility Requirements

1. This E-Covid-19 Support web application is very good supported software application.
2. Web application is helpful for people in this covid-19 pandemic situation.
3. Every patient can easily check get all the services he could need.

2.6 Security Requirements

2.6.1 Access Requirements

1. Only for registered user (doctor, patient, plasma donor, and admin) can access this system.
2. Guest is not allowed to have all features inside the system.

2.6.2 Integrity Requirements

1. This web application can't give access other user who is a guest.

2.7 Usability and Human Interaction Requirements

These requirements define how to meet the physical needs of the meant users of this web application.

2.7.1 Ease of Use Requirements

The system is easy to use and can understand easily by the system users.

UH-01	The system must be usable for doctors, patients, plasma donors and admin with all associate stakeholders.
Description	The system specify the several possible features for the users.
Stakeholders	Doctors, Patients, Plasma Donors & Admin.

2.7.2 Understand Ability and Politeness Requirements

This section describes about more requirements of this web application.

UH-02	The features of the E-Covid-19 Support.
Description	The system can more efficiently ease of use with all added features. The system has more understanding ability for all users. The system will not use any term which is not specified in this web application.
Stakeholders	System Admin.

2.7.3 Accessibility Requirements

There is no access requirements exclude those which have been listed in the below:

- AR-1: Log in as a Admin
- AR-2: Log in as a Doctor
- AR-3: Log in as a Patient
- AR-4: Log in as a Plasma Donor
- AR-5: Log out as Admin
- AR-6: Log out as Doctor
- AR-7: Log out as Patient
- AR-8: Log out as Plasma Donor

To get access into the system or to get specific module, this system will provide a control. In order to prevent anyone to exploit stolen all users details and passwords, for that all passwords of user must be encrypted in the hash process.

2.7.4 User Documentation Requirements

UH-03	The system developer documentation.
Description	To develop this system we have specified all the requirements of user's documentation. All the teams are involved in this project documentation with details.
Stakeholders	System Developer

2.8 Look and Feel Requirements

The look and feel requirements of E-Covid-19 Support describe the all desired spirit the mood, the style of the product's appearance. These all requirements specify by the intention of the appearance which are not detailed design of user interfaces.

2.8.1 Appearance Requirements

It should be clear to the system admin, doctors, patient and plasma donor needs to be filled and which can be left blank in this system.

LF-01	Labels of mandatory fields must be bold.
Description	Label of all mandatory fields must be bold so that we can identify them easily.
Stakeholders	Doctors, Patient, Plasma Donors & Admin.

2.8.2 Style Requirements

We will provide user interface of this system. This requirement not only define the importance to use a cascading style sheet (css) but also all the requirements regarding the css's content as well as css design like bootstrap, here I use bootstrap 4.

LF-02	The look and feel of this system must be controllable using a style sheets.
Description	The styling of this web application UI will be defined by using css, JS, Bootstrap 4.
Stakeholders	System Admin

2.9 Operational and Environmental Requirements

This requirement will focus how easily user can use this system, including all interfaces and interoperability. This requirement established that how much properly and under which conditions this system can be performed.

2.9.1 Expected Physical Requirements

There are no specific expected physical requirements for E-Covid-19 Support.

2.9.2 The Requirements for Interfacing with Adjacent System

There is no specific interface with adjacent system requirements for E-Covid-19 Support.

2.9.3 Release Requirements

There is no specific release requirement but in it described briefly the project schedule.

2.10 Legal Requirements

These requirements consider any violence of rules and regulations that which rules must have to maintain for system users. In this system it has no complain requirement.

CHAPTER-03

SYSTEM ANALYSIS

3.1 Use case Diagram

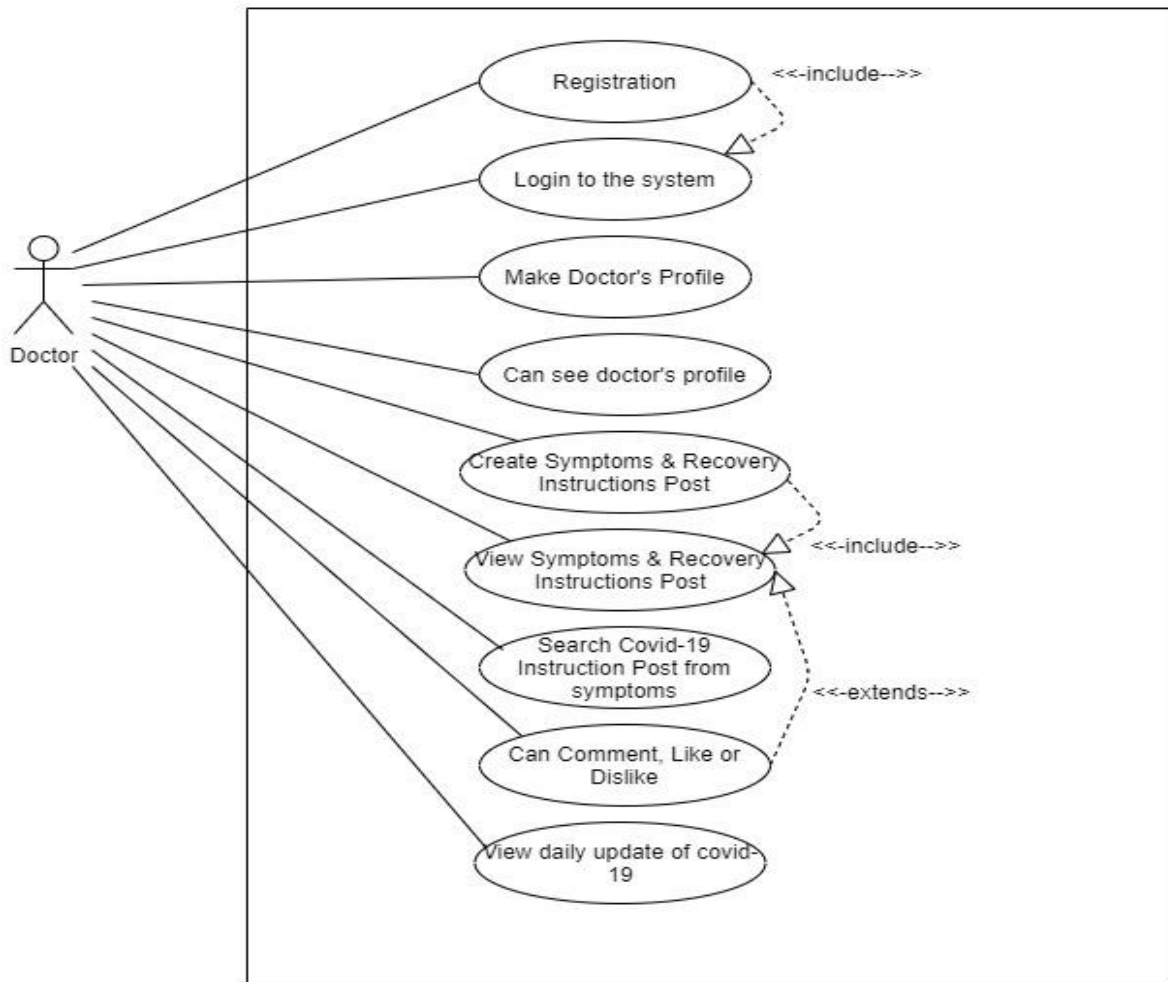


Figure 3.1.1: Use Case For Doctor

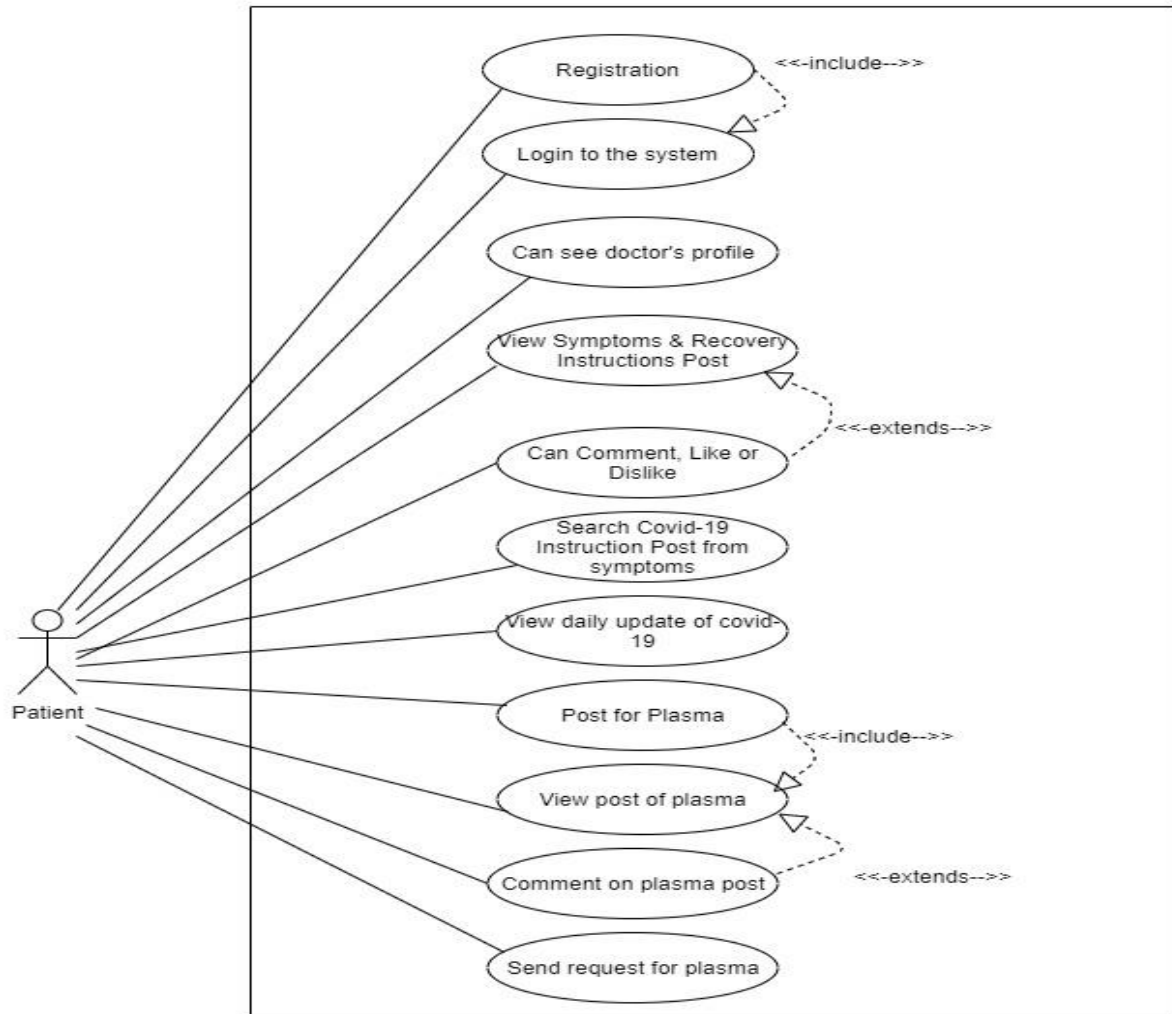


Figure 3.1.2: Use Case For Patient

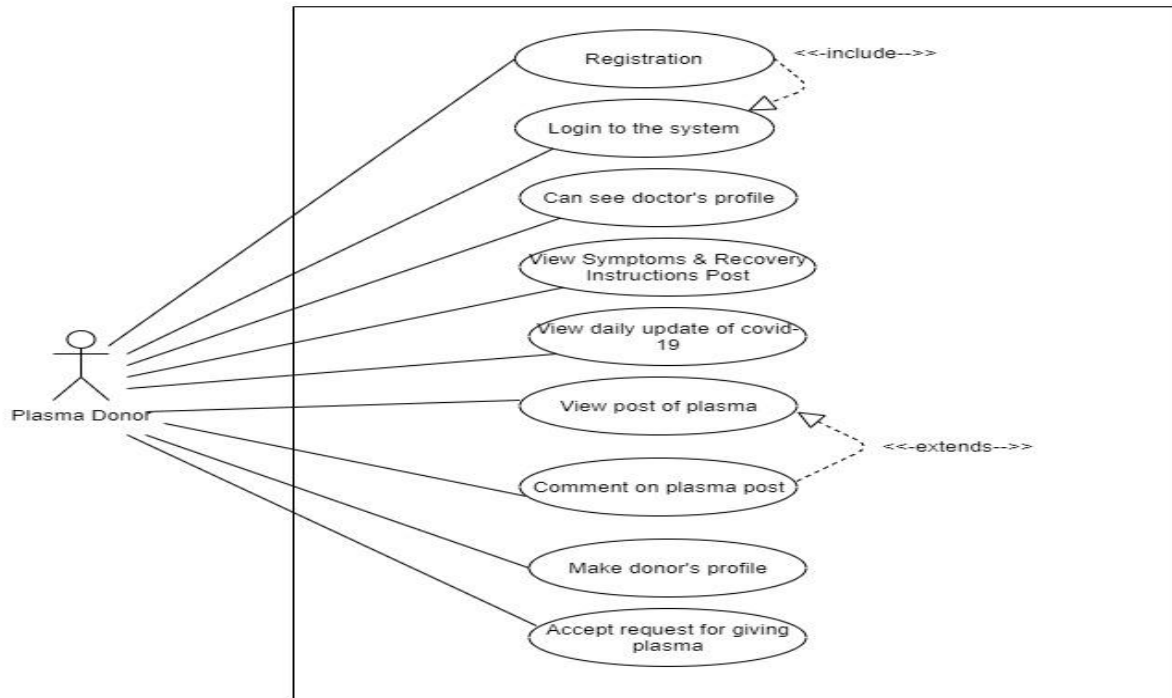


Figure 3.1.3: Use Case For Plasma Donor

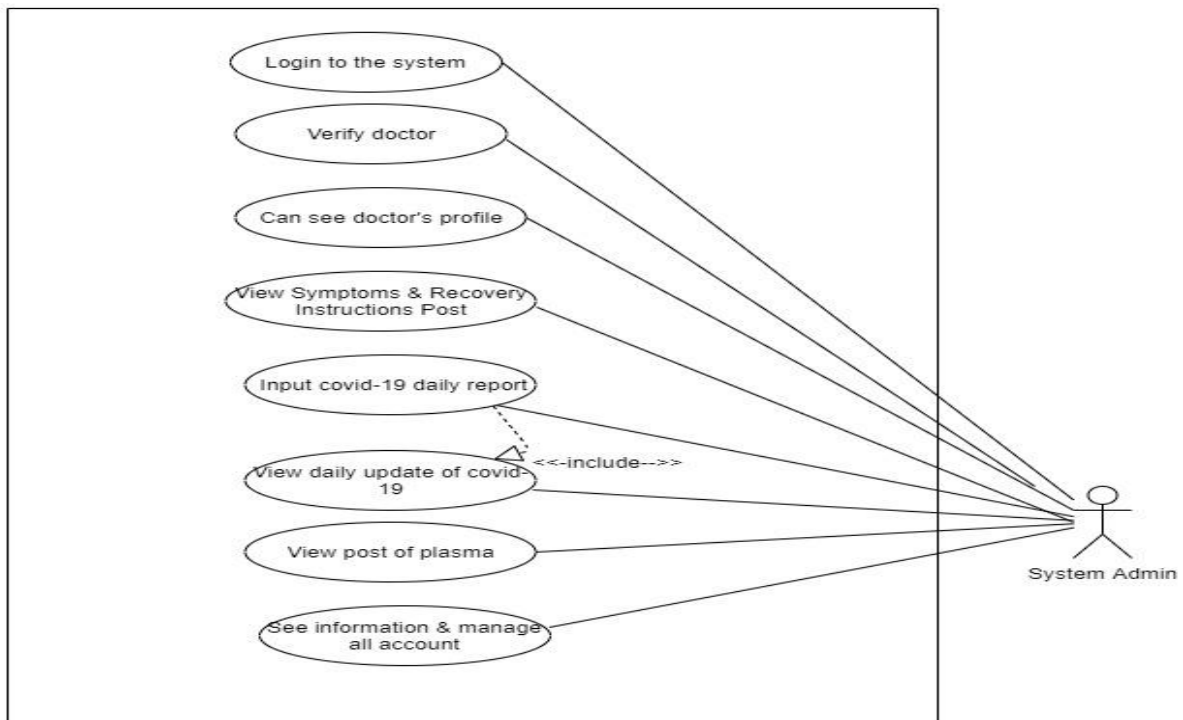


Figure 3.1.4: Use Case For Admin

3.2 Use Case Description

Use Case Name	Registration
Description	Doctor, Patient & Plasma Donor have to register for turn into the system.
Actor	Doctor, Patient & Plasma Donor
Precondition	User Should remain in registration page.
Trigger	Click the “Registration” link.
Flow of event	<ul style="list-style-type: none">• Have to input Name, email , password on the text field.• Write these and click the registration button.
Post Condition	User Account will be created.

Use Case Name	Login & Logout
Description	To get access into the system stakeholders must login and they can logout too.
Actor	Doctor , Patient, Plasma Donor & Admin
Precondition	User Should remain in login page.
Trigger	Click the “login” link.
Flow of event	<ul style="list-style-type: none">• Have to input email & password on the text field.• Write these and click the login button.
Post Condition	User logged into the system.

Use Case Name	Make Doctor’s Profile
Description	Doctor have to make their profile with professional details.
Actor	Doctor
Precondition	Login
Trigger	Click the “Add my profile” link.
Flow of event	<ul style="list-style-type: none">• Have to input all details on the text fields.• Write these and click the add profile button.
Post Condition	Store all details in doctor profile.

Use Case Name	See doctor’s profile
Description	All Stakeholder are able to see doctor’s profile after login in system.
Actor	Doctor , Patient, Plasma Donor & Admin
Precondition	Login
Trigger	Click the “Doctor List” link.
Flow of event	<ul style="list-style-type: none">• Select the doctor list link.• View every doctor profile whoever user want to see in details.
Post Condition	Able to see doctor list with details.

Use Case Name	Verify doctor
Description	Admin will verify doctor after his registration and give all access to doctor's system.
Actor	Admin
Precondition	Login
Trigger	Click the "Verify Doctor List" link.
Flow of event	<ul style="list-style-type: none"> • Select the verify doctor list link. • View registered doctor details with his doctor code and confirm him as a verified doctor.
Post Condition	Doctor able to see all type of access for doctor after his verification.

Use Case Name	Create symptoms & recovery instruction
Description	Only doctor can create this.
Actor	Doctor
Precondition	Login
Trigger	Click the "Add Symptoms & Recovery Instruction" link.
Flow of event	<ul style="list-style-type: none"> • Have to input detail on the text field. • Write these and click the save button.
Post Condition	Store in Recovery Instruction database.

Use Case Name	View symptoms & recovery instruction
Description	All stakeholders can see this.
Actor	Doctor , Patient, Plasma Donor & Admin
Precondition	Login
Trigger	Click the "Recovery Instructions" link.
Flow of event	<ul style="list-style-type: none"> • Click on post. • Will see a list.
Post Condition	Able to see symptoms & recovery instruction.

Use Case Name	Can comment on symptoms & recovery instruction
Description	Doctor & Patient both can comment.
Actor	Doctor & Patient
Precondition	Login
Trigger	Click the "Comment" link.
Flow of event	<ul style="list-style-type: none"> • Have to input detail on the text field. • Write these and click the comment button.
Post Condition	Comment will be posted.

Use Case Name	Search recovery instruction from symptoms
Description	Only patient can do this.
Actor	Patient
Precondition	Login
Trigger	Click the “Search From Symptoms” link.
Flow of event	<ul style="list-style-type: none"> • Have to input detail on the text field. • Write these and click go button.
Post Condition	Able to see sickness information in details.

Use Case Name	Create plasma post
Description	Only patient can create this.
Actor	Patient
Precondition	Login
Trigger	Click the “Add Plasma Post” link.
Flow of event	<ul style="list-style-type: none"> • Have to input detail on the text field. • Write these and click the save button.
Post Condition	Store in Plasma post database.

Use Case Name	View plasma post
Description	Patient, Plasma Donor & Admin can see this.
Actor	Patient, Plasma Donor & Admin
Precondition	Login
Trigger	Click the “Recovery Instructions” link.
Flow of event	<ul style="list-style-type: none"> • Click on Plasma post. • Will see a list.
Post Condition	Able to see plasma posts.

Use Case Name	Can comment on plasma post
Description	Patient & Plasma Donor both can comment.
Actor	Patient & Plasma Donor
Precondition	Login
Trigger	Click the “Comment” link.
Flow of event	<ul style="list-style-type: none"> • Have to input detail on the text field. • Write these and click the comment button.
Post Condition	Comment will be posted on plasma post.

Use Case Name	Make Plasma Donor's Profile
Description	Plasma Donor have to make their profile with all details.
Actor	Plasma Donor
Precondition	Login
Trigger	Click the "Add plasma donor profile" link.
Flow of event	<ul style="list-style-type: none"> • Have to input all details on the text fields. • Write these and click the add profile button.
Post Condition	Store all details in plasma donor profile.

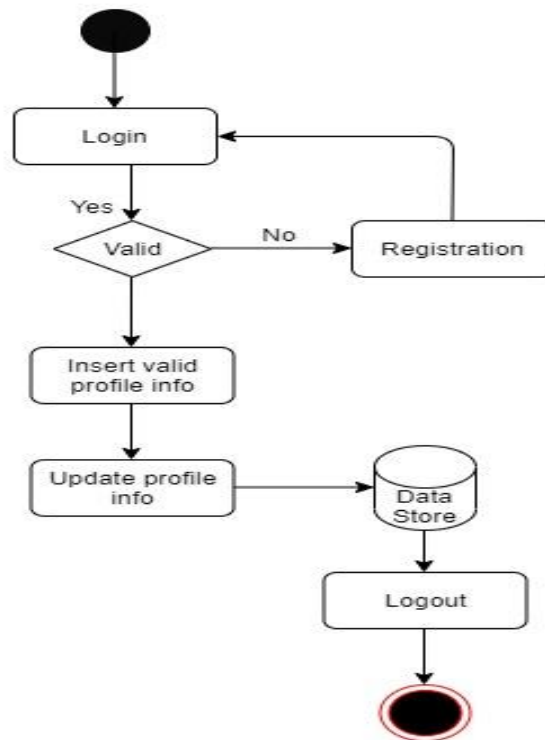
Use Case Name	Send request to plasma donor
Description	Patient can send request to plasma donor for donate plasma.
Actor	Patient
Precondition	Login
Trigger	Click the "Send request" link.
Flow of event	<ul style="list-style-type: none"> • Have to click on send request button.
Post Condition	Store all details in send request.

Use Case Name	Accept request
Description	Plasma Donor have accept request to show his contact number which is send by patient.
Actor	Plasma Donor
Precondition	Login
Trigger	Click the "Accept request" link.
Flow of event	<ul style="list-style-type: none"> • Have to click on accept request button.
Post Condition	Store all details in accept request.

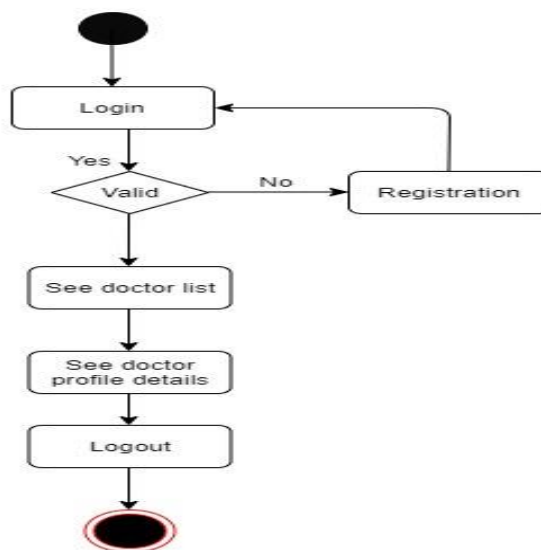
Use Case Name	Input Covid-19 Daily Report
Description	Admin will input daily effected people of covid-19.
Actor	Admin.
Precondition	Login
Trigger	Click the "Add Covid-19 Report" link.
Flow of event	<ul style="list-style-type: none"> • Have to input all details on the text fields. • Write these and click the add report.
Post Condition	Store all details in covid report.

3.3 Activity Diagram

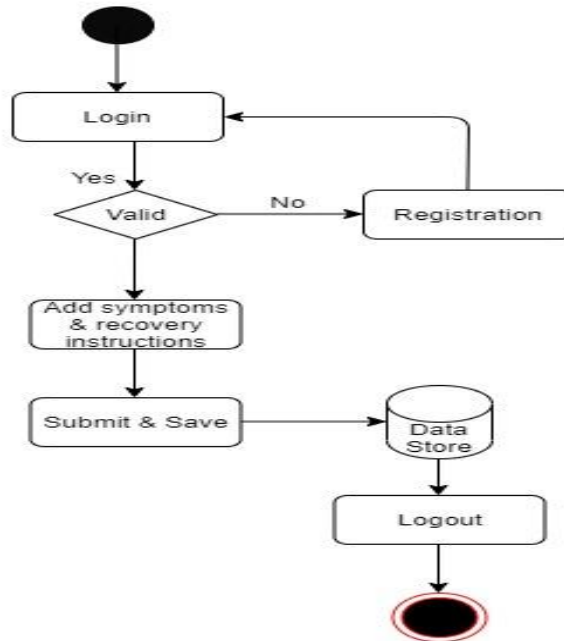
3.3.1 Make Doctor's profile



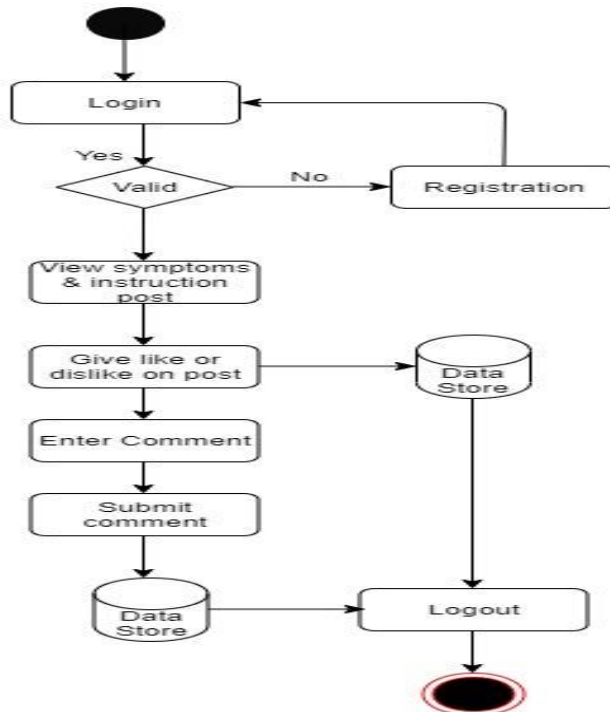
3.3.2 View Doctor's profile



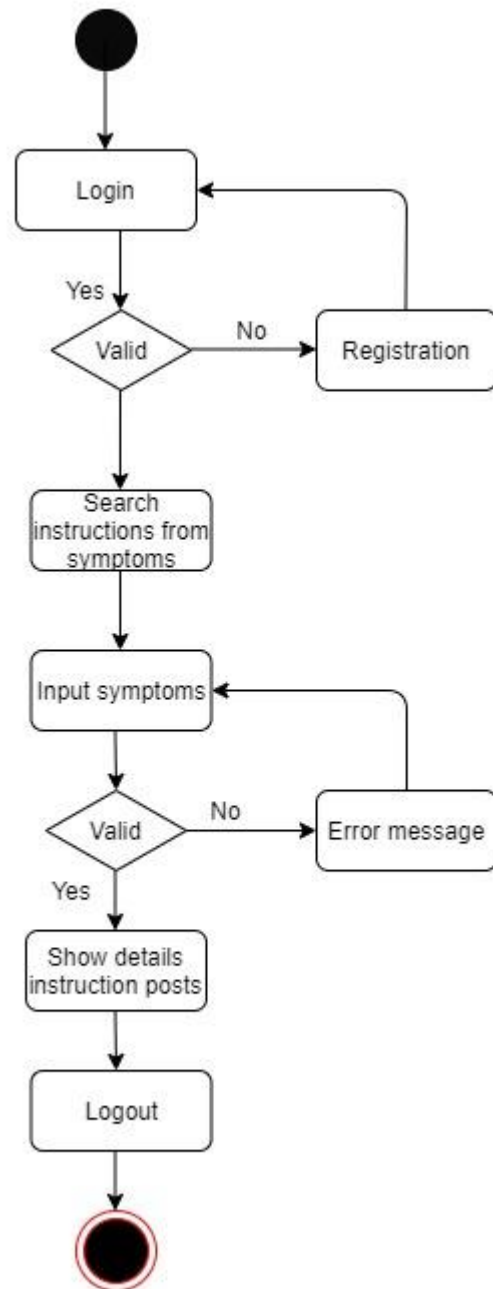
3.3.3 Create Symptoms & Recovery Instruction Post



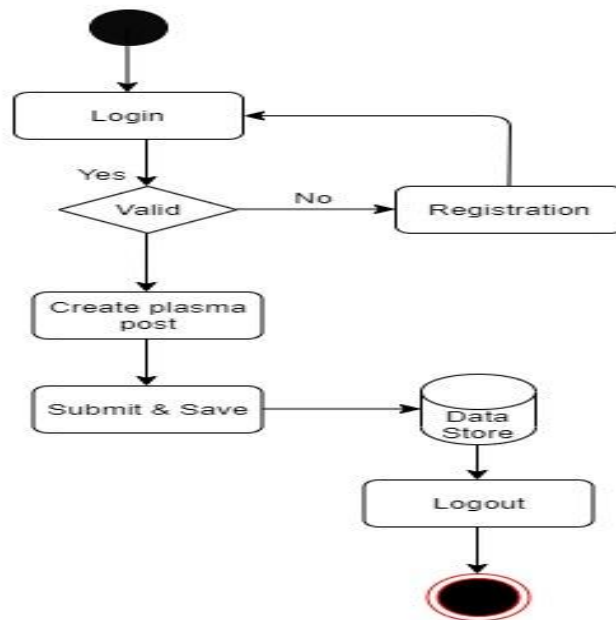
3.3.4 Can Comment, Like Or Dislike on doctor's post



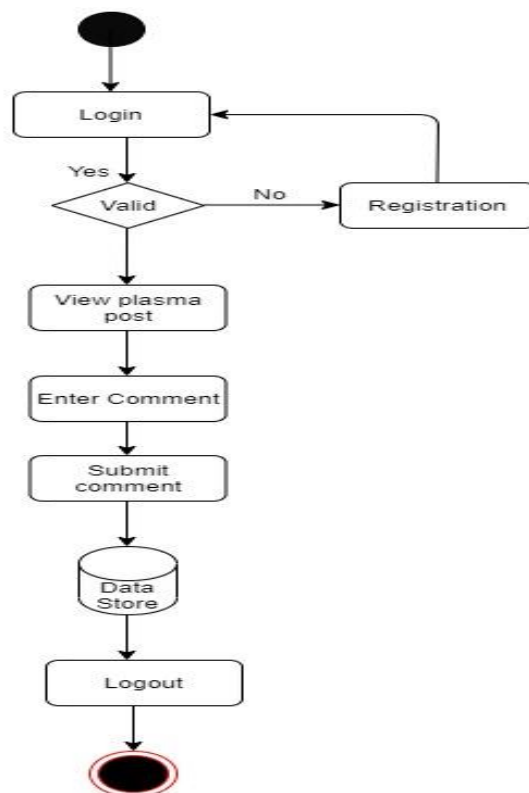
3.3.5 Search Recovery Instruction From Symptoms



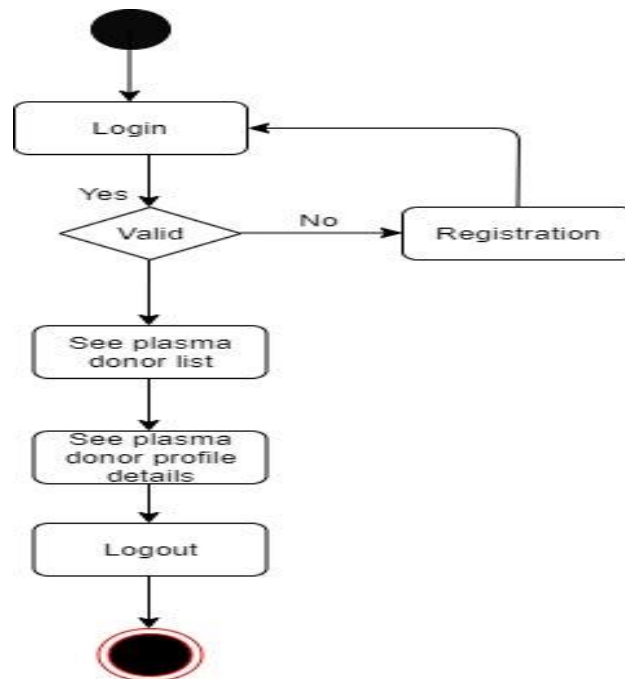
3.3.6 Create Post for Plasma



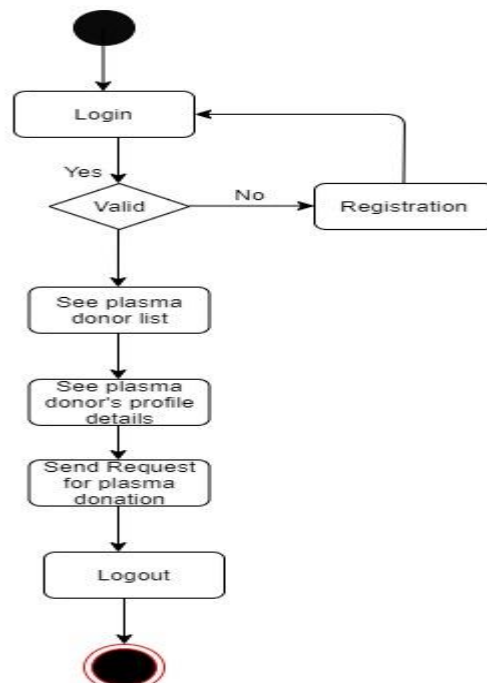
3.3.7 Comment on Plasma Post



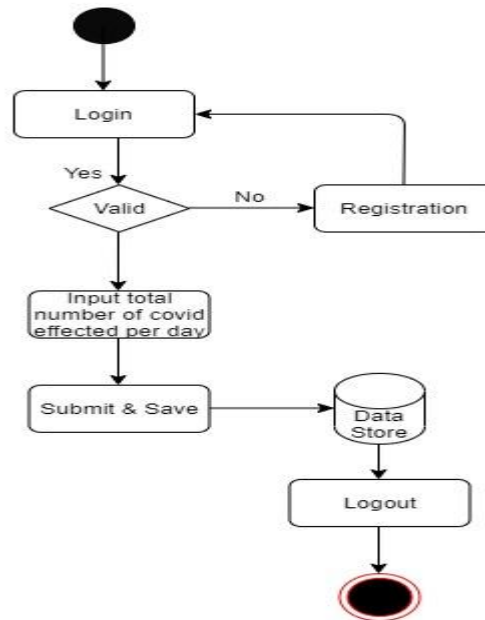
3.3.8 View Plasma Donor's profile



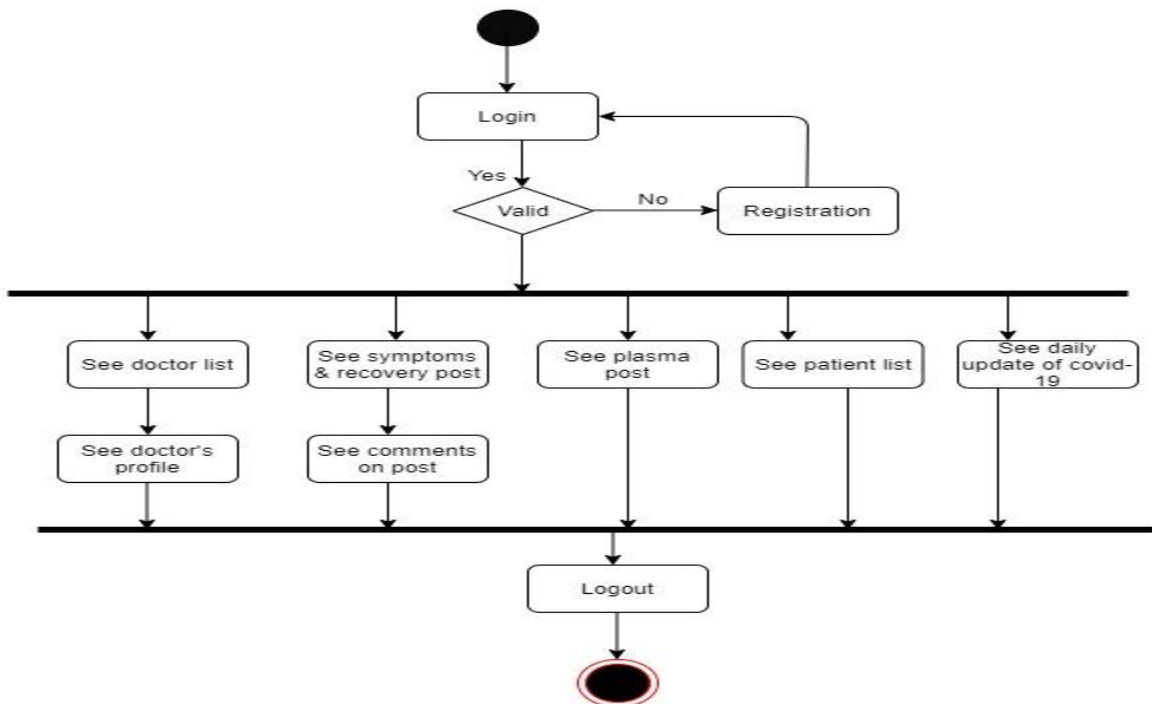
3.3.9 Accept Plasma Request



3.3.10 Input Covid-19 Daily Report



3.3.11 See all information

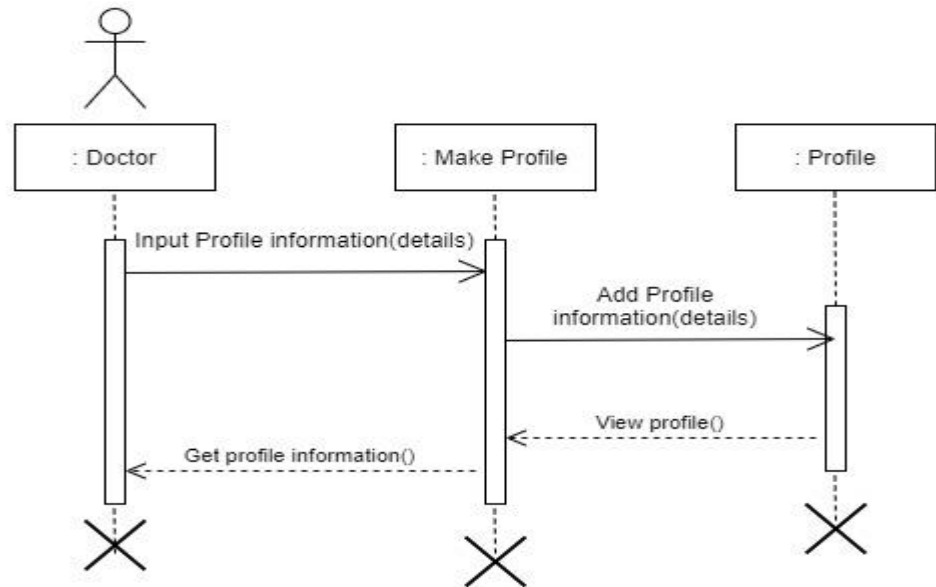


CHAPTER-04

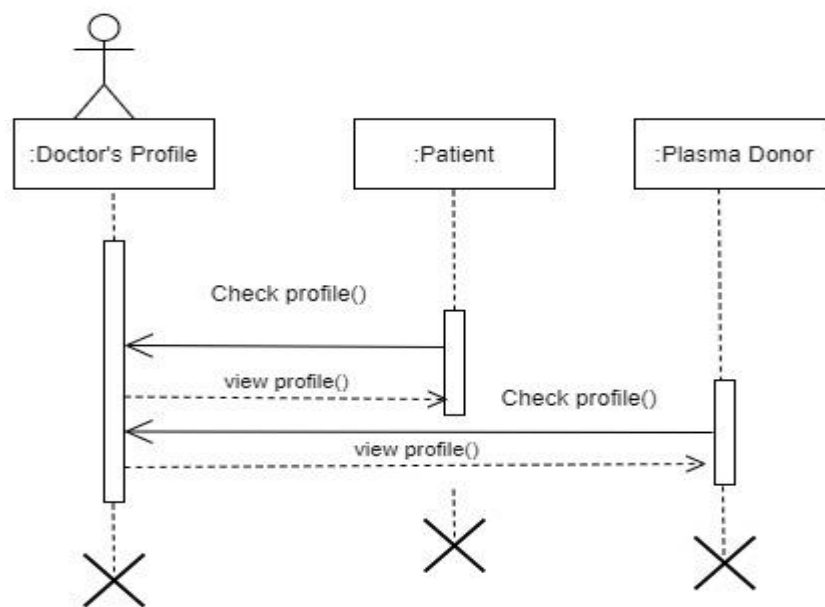
System Design Specification

4.1 Sequence Diagram

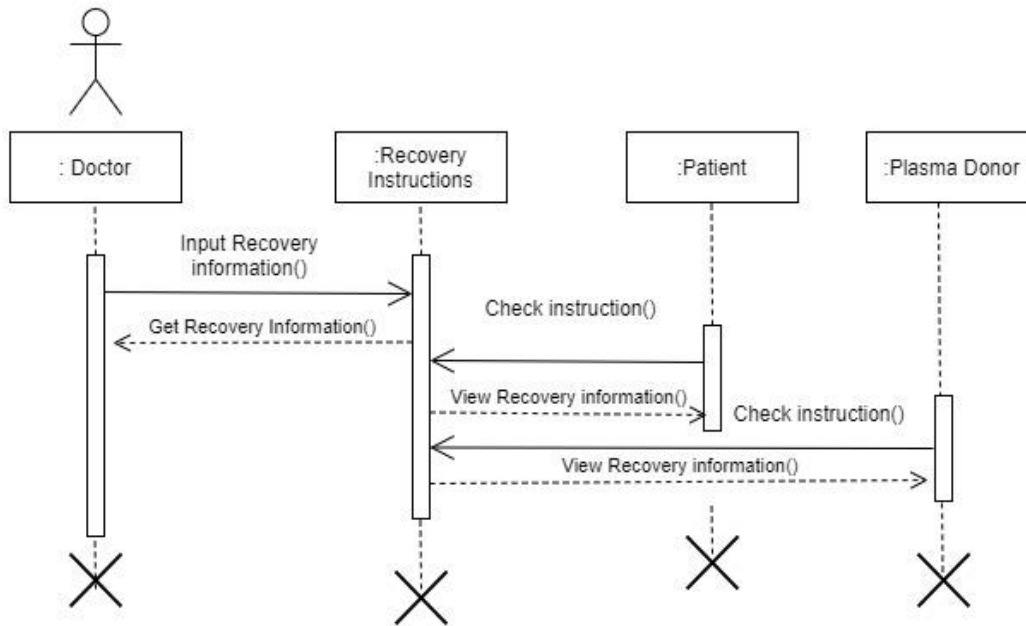
4.1.1 Make Doctor's profile



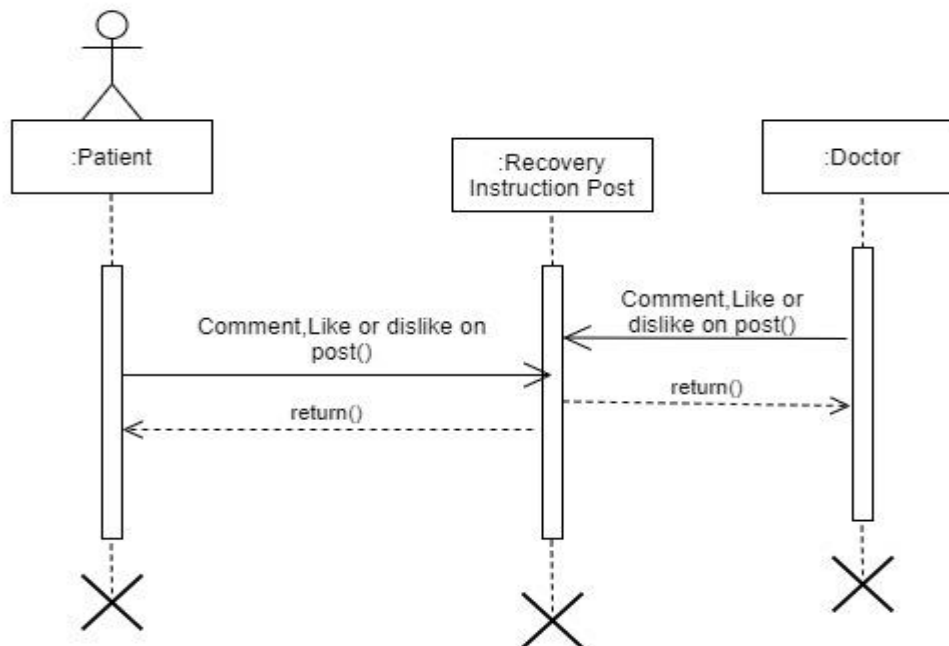
4.1.2 View Doctor's profile



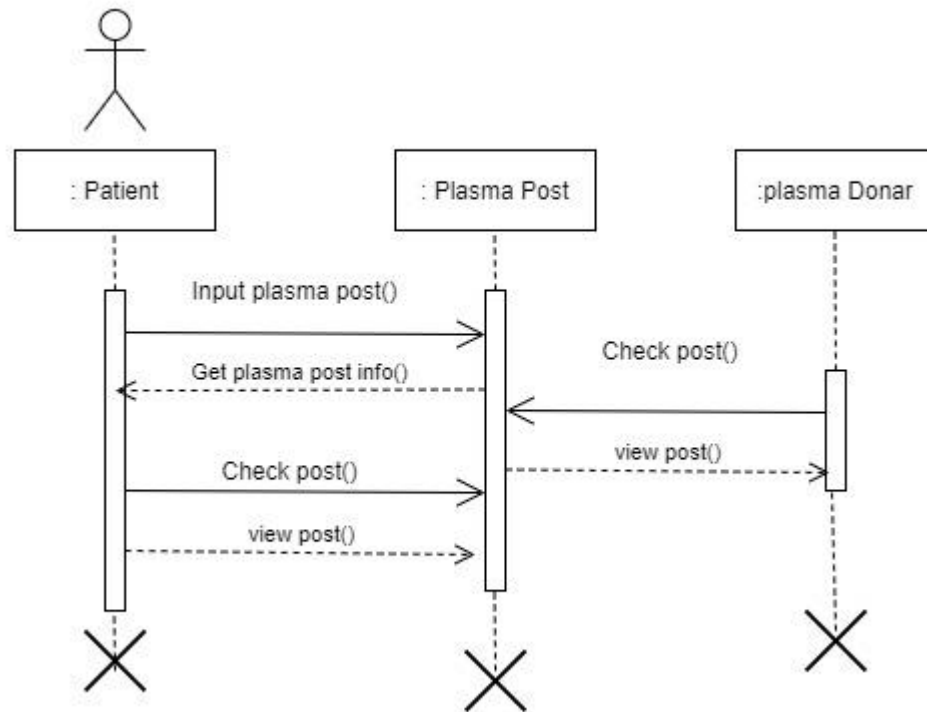
4.1.3 Create Symptoms & Recovery Instruction Post



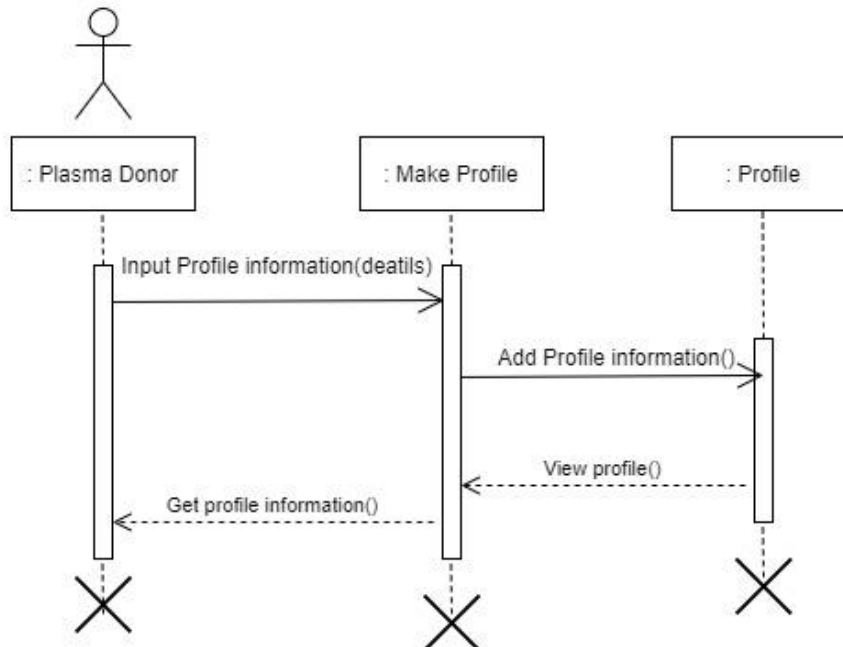
4.1.4 Comment on Symptoms & Recovery Instruction Post



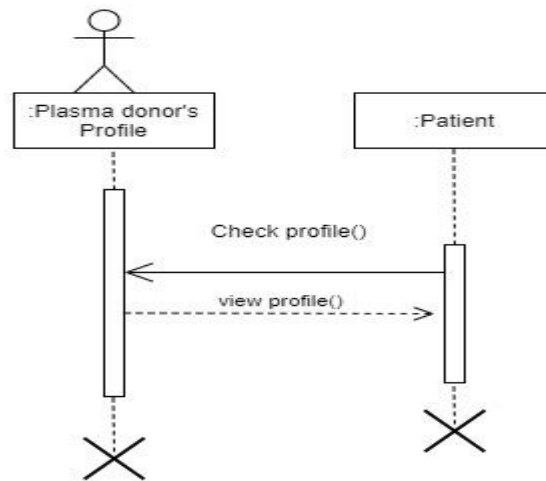
4.1.5 View Plasma Post



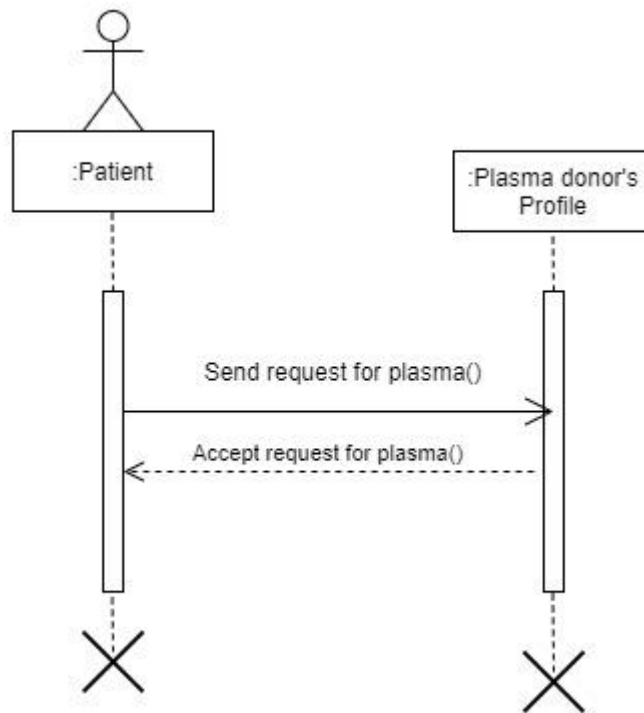
4.1.6 Make Plasma Donor's profile



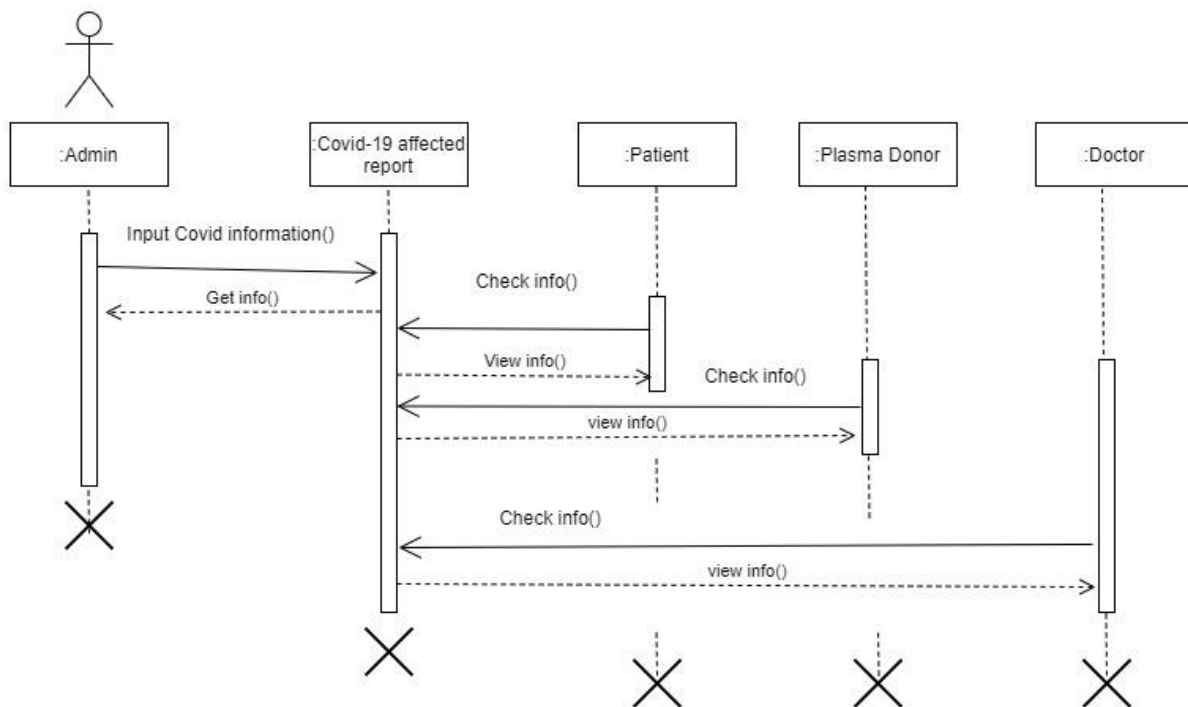
4.1.7 View Plasma Donor's Profile



4.1.8 Send Request to Plasma Donor



4.1.9 View Daily Covid-19 Report



4.2 E-R Diagram



Figure 4.2: ER Diagram

4.3 Class Diagram

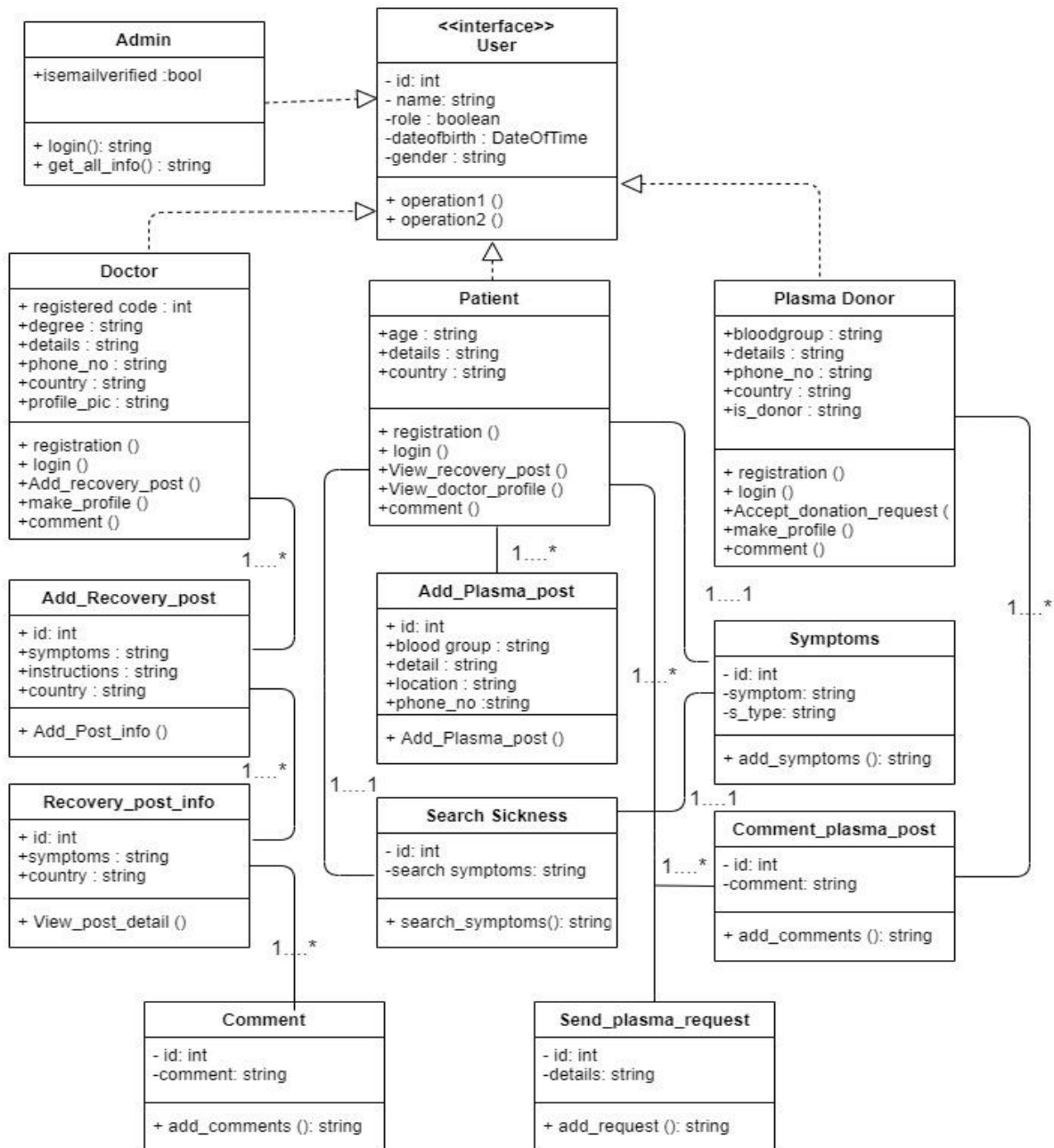


Figure 4.3: Class Diagram

4.4 Development Tools and Technology

4.4.1 User Interface Technology

4.4.1.1: HTML, HTML5

4.4.1.2: CSS3, CSS

4.4.1.3: Bootstraps-4

4.4.1.3: JavaScript, Font Awesome- 5

4.4.2 Implementation Tools and Platforms

4.4.2.1: Laravel version - 5.8

4.4.2.2: Xampp version -3.2.4

4.4.2.3: Visual Studio Code

CHAPTER-05

System Testing

5.1 Testing Features

5.1.1 Feature to be tested

1. Registration
2. Login
3. Can see doctor's profile
4. Create symptoms & recovery post
5. View symptoms & recovery post
6. Search covid-19 instruction from symptoms
7. Input covid-19 daily report
8. View covid-19 daily report
9. Post for plasma
10. View post of plasma
11. Send request for plasma
12. Accept request for giving plasma

5.1.2 Feature not to be tested

1. Feedbacks

5.2 Testing Strategies

5.2.1 Test Approach

1. The Complete system has been tested manually.
2. System testing based on Doctor, Patient and Plasma Donor.
3. This system test on doctor activities, patient activities and plasma donor activities.

5.2.2 Pass or Fail Criteria

1. Component Pass/Fail Criteria: - The test will be passed the cases when it will meet the object design requirements.
2. Integration Pass/Fail Criteria: - The test will be failed the cases when it will not meet the object architecture requirements.

5.2.3 Suspension and Resumption

1. Regression Testing: - After each changes, this system will be worked perfectly.
2. Database Change: - If you change the database name of the system then the system will not support accurately.
3. System Design Change: - After changes in designing part the system will not work accurately.

5.2.4 Testing Schedule

In below I am showing the testing schedule table where I discuss about my project work plans with it's estimation time.

Table 5.1: Testing Schedule

Test Phase	Time
Test Plan	1 Week
Test Specification	2 Week
Test Specification Team	1 Week
Component Test	3 Week
Integration Test	2 Week
System Testing	4 Week

5.3 Test Cases

5.3.1 Test Case 1 (User Login)

Table 5.2: Test Case 1 (User Login)

Test Case: - 01	Test Case Name: Login
System: User Login	Subsystem: N/A
Designed By: -Kamrun Nahar Liza (172-35-2144)	Design Date: 05-03-2021
Execute By: - Kamrun Nahar Liza	Execution date: - 15-03-2021

Table 5.3: User Login

Step	Action	Expected System response	Pass/Fail	Comment
1	When system user fills all login fields and click on login button.	If user don't enter email id then show required email id.	Pass	Email Field are required
2	When user enters invalid email format	The system will show this field needs valid email format required.	Pass	The valid email format needs to login

5.3.2 Test Case 2 (Create Symptoms & Recovery Post)

Table 5.4: Test Case 2 (Create symptoms & recovery post)

Test Case: - 02	Test Case Name: Create symptoms & recovery post
System: Add symptoms & recovery post	Subsystem: N/A
Designed By: -Kamrun Nahar Liza (172-35-2144)	Design Date: 15-03-2021
Execute By: - Kamrun Nahar Liza	Execution date: - 25-03-2021

Table 5.5: Create symptoms & recovery post

Step	Action	Expected System response	Pass/Fail	Comment
1	When doctor fills up the symptoms field and click post button	If doctor missing any field then this system shows required field.	Pass	All fields are required here.
2	When doctor fills up the recovery instruction field and click post button	If doctor missing any field then this system shows required field.	Pass	All fields are required here.

5.3.3 Test Case 3 (View doctor's profile)

Table 5.6: Test Case 3 (View doctor's profile)

Test Case: - 03	Test Case Name: View doctor's profile
System: View profile	Subsystem: N/A
Designed By: -Kamrun Nahar Liza (172-35-2144)	Design Date: 15-03-2021
Execute By: - Kamrun Nahar Liza	Execution date: - 25-03-2021

Table 5.7: View doctor's profile

Step	Action	Expected System response	Pass/Fail	Comment
1	When registered user click on doctor's list.	He will get all available registered doctor list.	Pass	Have to click on doctor's list.
2	When inside doctor's list for click on view profile of any specific doctor.	He will able to see that specific one doctor's details.	Pass	Have to click on view profile.

5.3.4 Test Case 4 (Create Plasma Post)

Table 5.8: Test Case 4 (Create plasma post)

Test Case: - 02	Test Case Name: Create plasma post
System: Create plasma post	Subsystem: N/A
Designed By: -Kamrun Nahar Liza (172-35-2144)	Design Date: 15-03-2021
Execute By: - Kamrun Nahar Liza	Execution date: - 25-03-2021

Table 5.9: Create plasma post

Step	Action	Expected System response	Pass/Fail	Comment
1	When patient fills all fields and click post button.	If patient missing any field then this system shows required field.	Pass	All fields are required here.
2	Patient need to add the country and location.	If patient missing any field then this system shows required field.	Pass	All fields are required here.

5.3.5 Test Case 4 (Input Covid-19 daily report)

Table 5.10: Test Case 5 (Input covid-19 daily report)

Test Case: - 05	Test Case Name: Input covid-19 daily report
System: Input covid-19 daily report	Subsystem: N/A
Designed By: -Kamrun Nahar Liza (172-35-2144)	Design Date: 15-03-2021
Execute By: - Kamrun Nahar Liza	Execution date: - 25-03-2021

Table 5.11: Input covid-19 daily report

Step	Action	Expected System response	Pass/Fail	Comment
1	When admin fills up all fields and click post button.	If admin missing any field then this system shows required field.	Pass	All fields are required here.
2	Admin need to add the total report.	If admin missing any field then this system shows required field.	Pass	All fields are required here.

5.3.6 Test Case 6 (Send request for plasma)

Table 5.12: Test Case 6 (Send request for plasma)

Test Case: - 06	Test Case Name: Send request for plasma
System: Send request	Subsystem: N/A
Designed By: -Kamrun Nahar Liza (172-35-2144)	Design Date: 15-03-2021
Execute By: - Kamrun Nahar Liza	Execution date: - 25-03-2021

Table 5.13: Send request for plasma

Step	Action	Expected System response	Pass/Fail	Comment
1	When patient click on plasma donors list button.	He will get all available registered plasma donors list.	Pass	Have to click on plasma donor's list.
2	When inside plasma donor's list for click on send request of any specific donor.	He will able to send request for donating plasma.	Pass	Have to click on send request.

5.4 Testing Environment (hardware/software requirements)

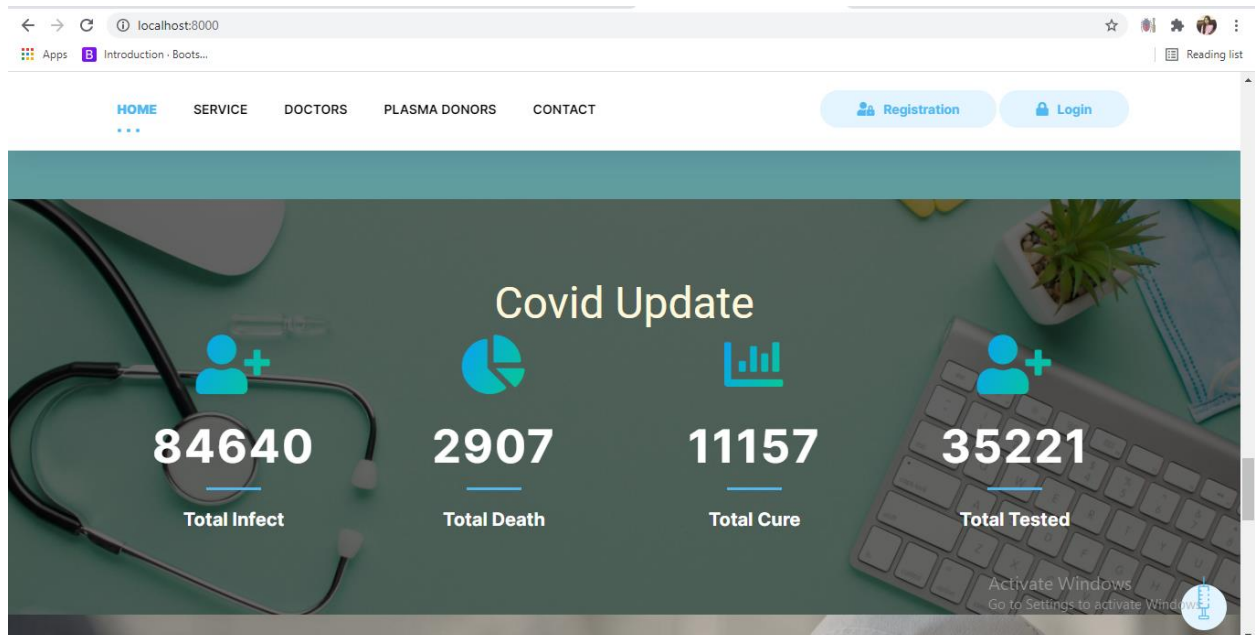
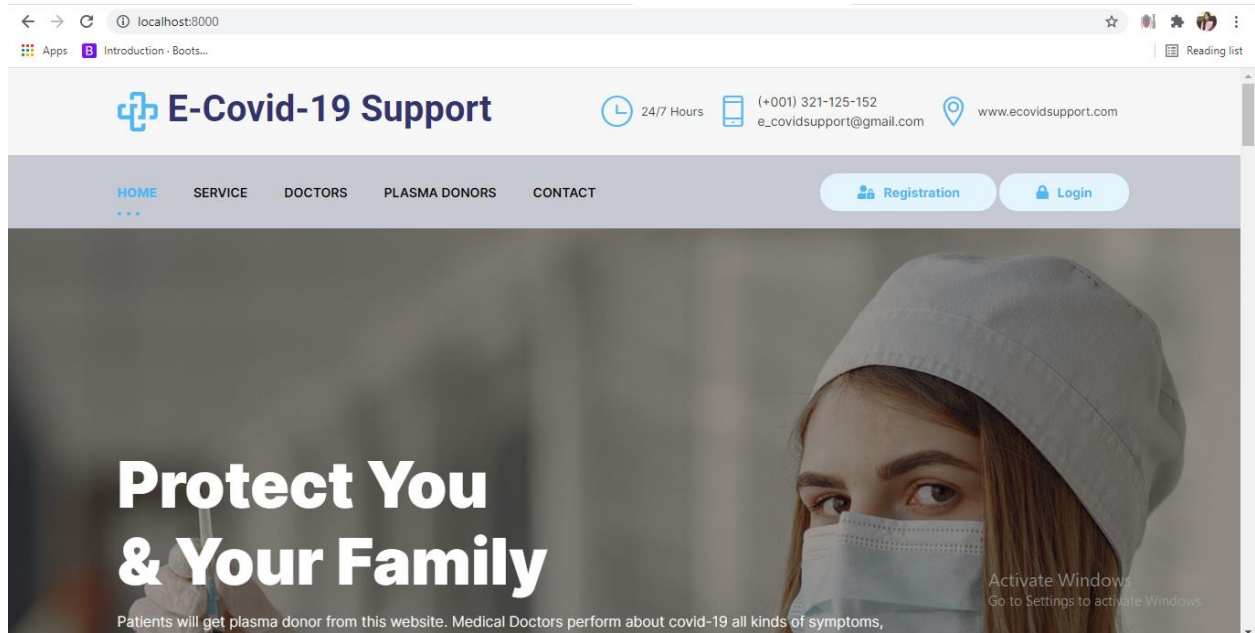
1. Browser: - Firefox, Google Chrome, Microsoft edge.
2. Core I3, Ram# 8GB, SSD#120GB

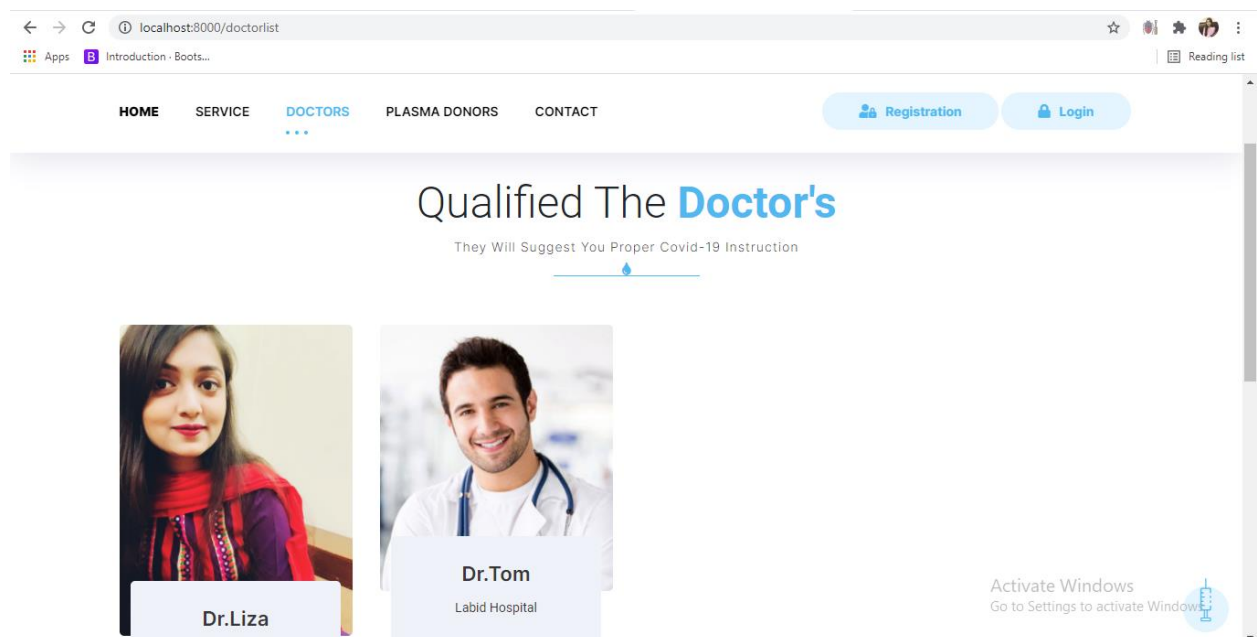
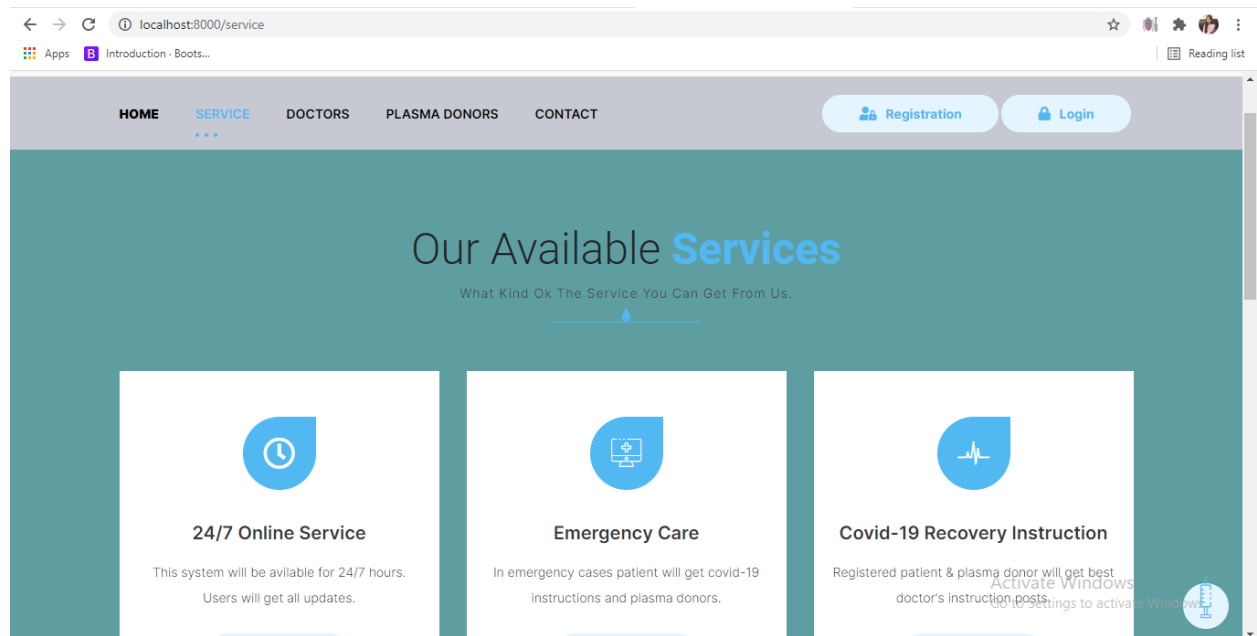
CHAPTER – 06

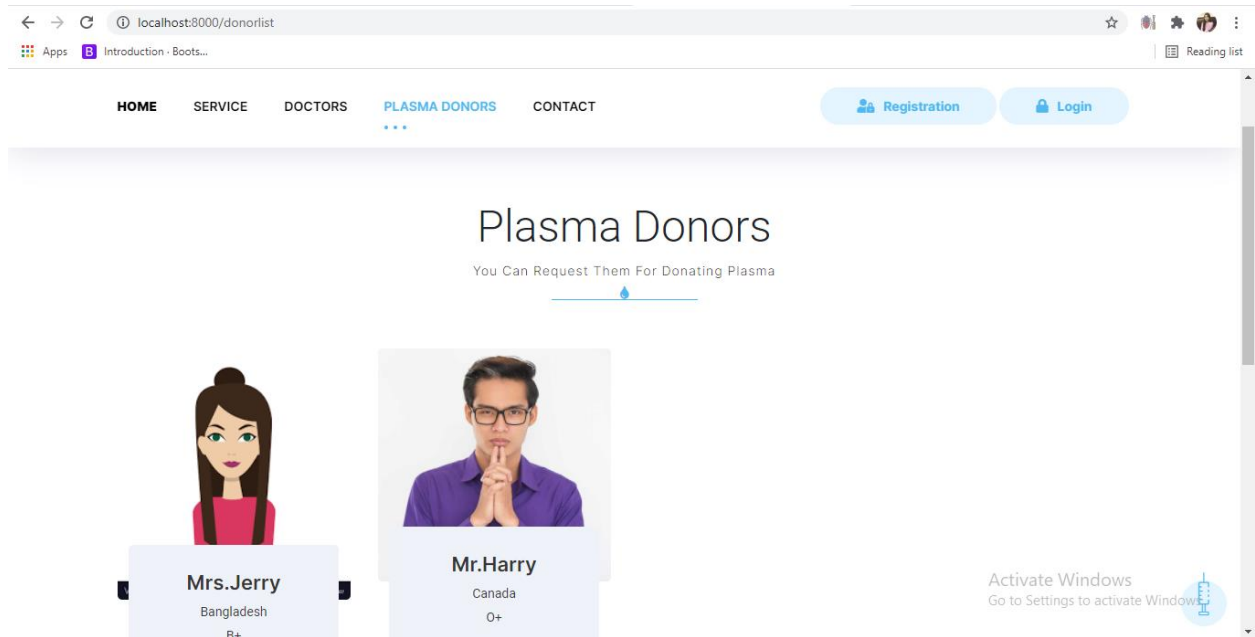
User Manual

6.1 User Manual

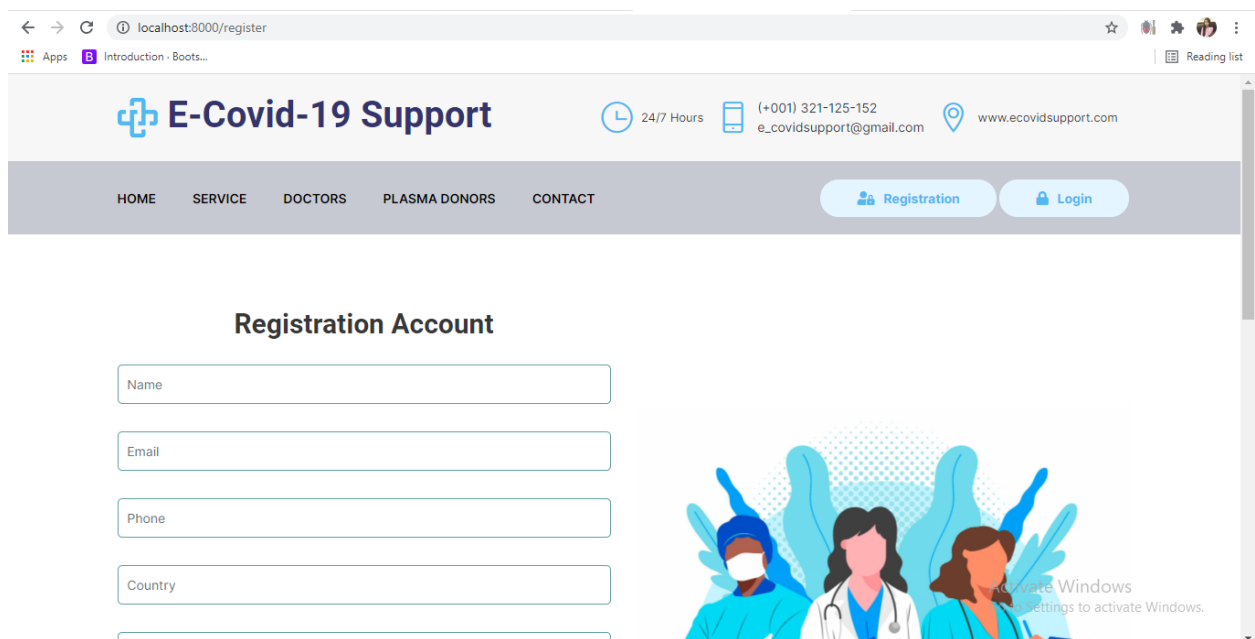
1. Go to website <http://localhost:8000>







2. Complete Registration (Fill up Name, Valid email, phone no, role, Password)



3. When a doctor will registered with his registered doctor code he will not get any access of doctor, after approving by admin he will get all access of doctor.

The screenshot shows the Admin interface of the E-Covid-19Support system. The top navigation bar is blue with the text "E- Covid-19Support" and "Admin" on the right. A sidebar on the left contains a menu with items: Dashboard, Covid Update, Verify Doctor, Plasma Post, Doctor List, Country List, Recovery Instructions, and Plasma Donor List. The main content area is titled "Doctor Verification List" and contains a table with the following data:

Name	Role	Email	Phone	Country	Registered Doctor Code	Status	Action
Dr. Liza	doctor	Liza@gmail.com	01771234567	Bangladesh	2345	Accepted	Pending
Dr. Tom	doctor	Tom@gmail.com	01798765432	Canada	4567	Accepted	Pending
Dr. Watson	doctor	Watson@gmail.com	0123456788	Bangladesh	9854	Pending	Accept Delete

At the bottom of the page, there is a copyright notice: "Copyright © 2020 E- Covid-19 Support. All rights reserved." and a watermark for "Activate Windows".

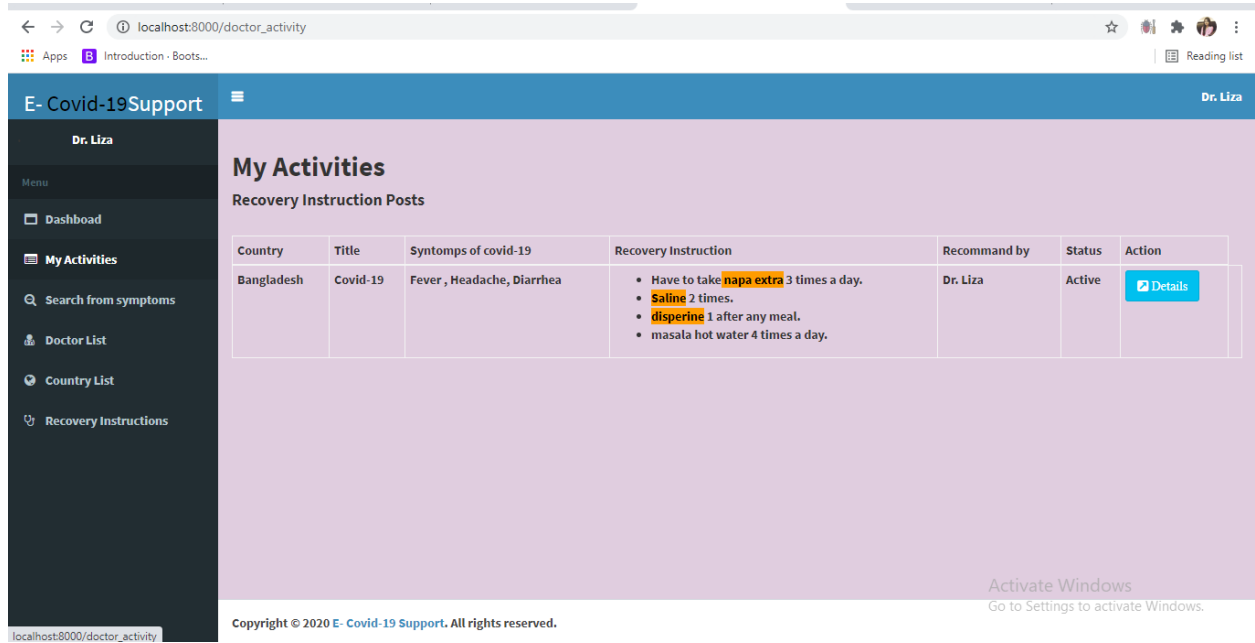
4. After login as approved doctor he will get all access and see covid-19 report on his dashboard.

The screenshot shows the Doctor interface of the E-Covid-19Support system. The top navigation bar is blue with the text "E- Covid-19Support" and "Dr. Liza" on the right. A sidebar on the left contains a menu with items: Dashboard, My Activities, Search from symptoms, Doctor List, Country List, and Recovery Instructions. The main content area is titled "Hello Dr. Liza" and "You are logged in Dr. Liza as doctor". It displays four cards with the following data:

24 Hours:	24 Hours:	24 Hours:	24 Hours:
1800	100	800	12345
Total: 128410793	Total: 2807311	Total: 103573977	Total: 228777777
New Infected	Death	Cure	Total Test

At the bottom of the page, there is a copyright notice: "Copyright © 2020 E- Covid-19 Support. All rights reserved." and a watermark for "Activate Windows".

5. Doctor can see his all activity from my activity option.



localhost:8000/doctor_activity

E- Covid-19Support Dr. Liza

Menu

- Dashboard
- My Activities**
- Search from symptoms
- Doctor List
- Country List
- Recovery Instructions

My Activities

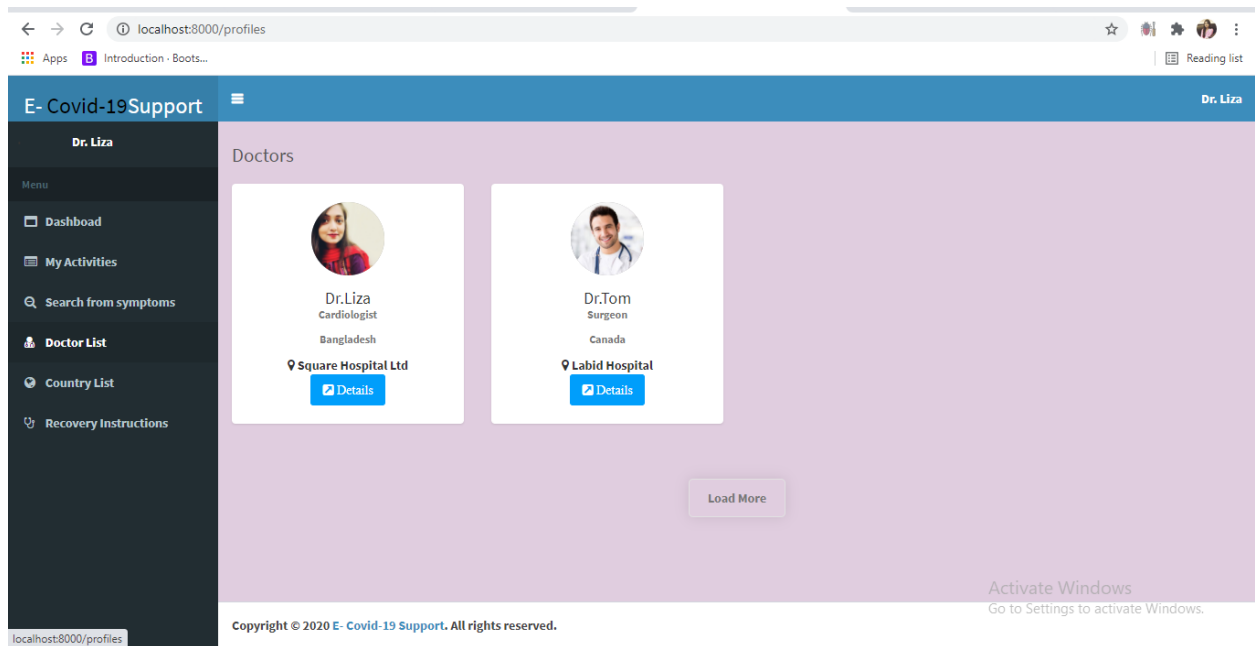
Recovery Instruction Posts

Country	Title	Syntomps of covid-19	Recovery Instruction	Recommend by	Status	Action
Bangladesh	Covid-19	Fever , Headache, Diarrhea	<ul style="list-style-type: none">Have to take napa extra 3 times a day.Saline 2 times.disperine 1 after any meal.masala hot water 4 times a day.	Dr. Liza	Active	Details

Activate Windows
Go to Settings to activate Windows.

Copyright © 2020 E- Covid-19 Support. All rights reserved.

6. User can see all doctor lists with details. Doctor can edit his own information.




localhost:8000/profiles

E- Covid-19Support Dr. Liza


Menu

- Dashboard
- My Activities
- Search from symptoms
- Doctor List**
- Country List
- Recovery Instructions

Doctors



Dr. Liza
Cardiologist
Bangladesh
Square Hospital Ltd
[Details](#)



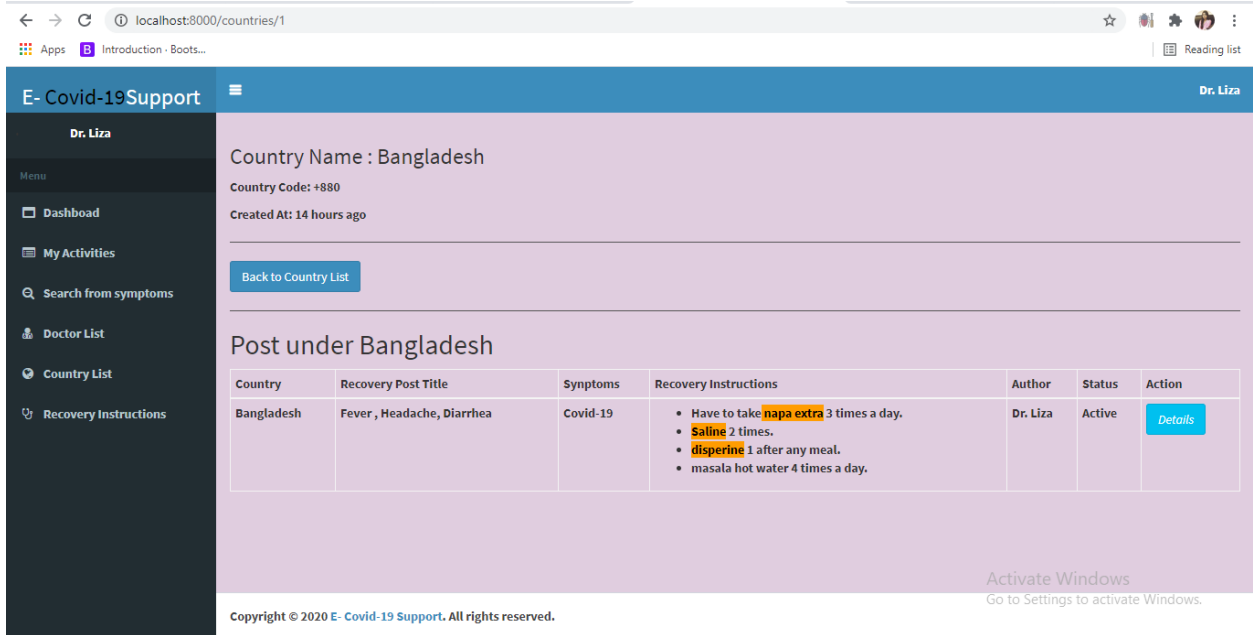
Dr. Tom
Surgeon
Canada
Labid Hospital
[Details](#)

[Load More](#)

Activate Windows
Go to Settings to activate Windows.

Copyright © 2020 E- Covid-19 Support. All rights reserved.

7. From country list registered user can see recovery posts under that country.

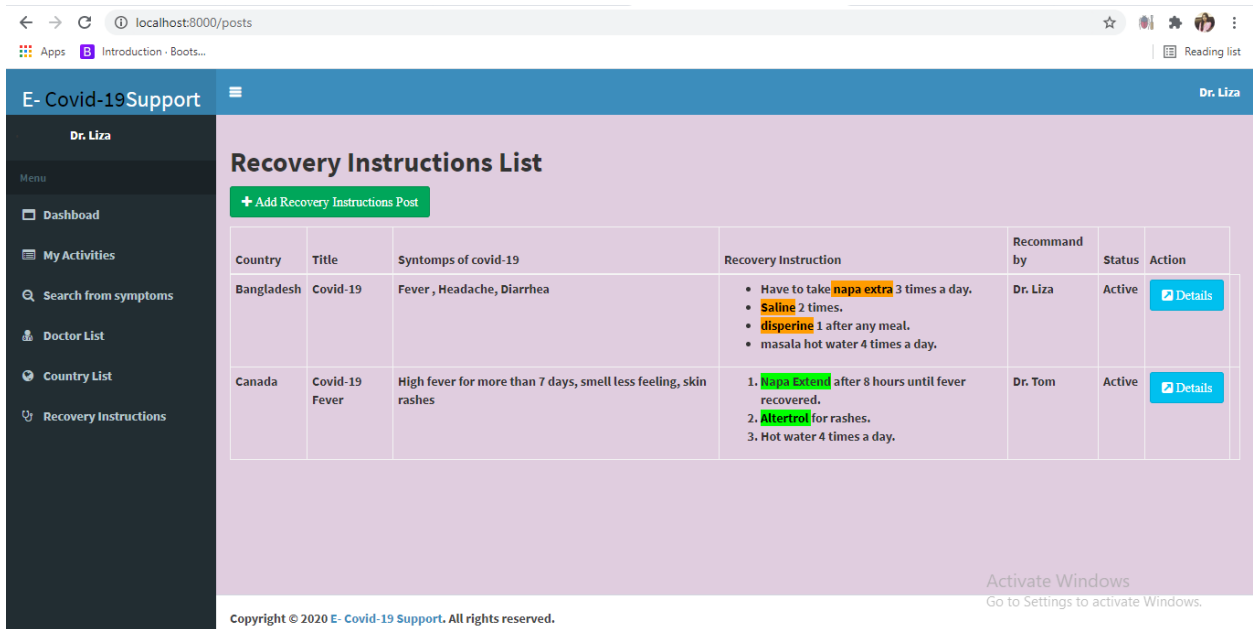


The screenshot shows the 'E- Covid-19Support' application interface. The user is logged in as 'Dr. Liza'. The page title is 'Country Name : Bangladesh'. Below the title, it shows 'Country Code: +880' and 'Created At: 14 hours ago'. There is a 'Back to Country List' button. The main section is titled 'Post under Bangladesh' and contains a table with the following data:

Country	Recovery Post Title	Symptoms	Recovery Instructions	Author	Status	Action
Bangladesh	Fever , Headache, Diarrhea	Covid-19	<ul style="list-style-type: none">Have to take napa extra 3 times a day.Saline 2 times.disperine 1 after any meal.masala hot water 4 times a day.	Dr. Liza	Active	Details

At the bottom of the page, there is a copyright notice: 'Copyright © 2020 E- Covid-19 Support. All rights reserved.' and an 'Activate Windows' watermark.

8. Doctor can add recovery instruction post, other users can see details only.

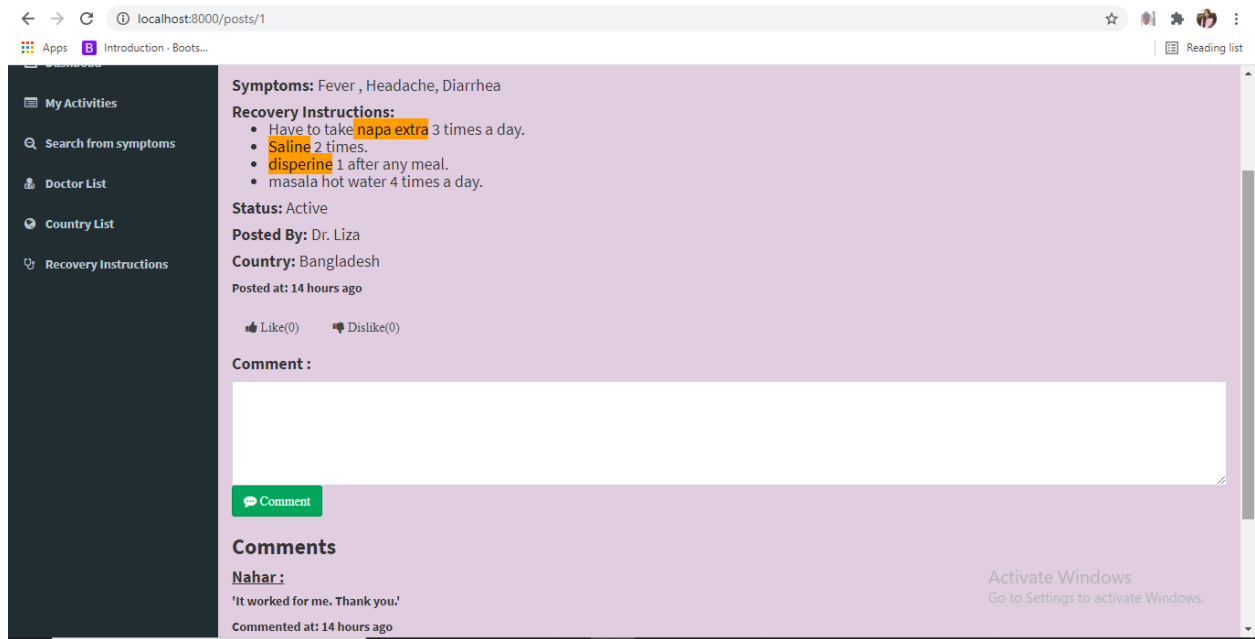


The screenshot shows the 'E- Covid-19Support' application interface. The user is logged in as 'Dr. Liza'. The page title is 'Recovery Instructions List'. Below the title, there is a green button '+ Add Recovery Instructions Post'. The main section contains a table with the following data:

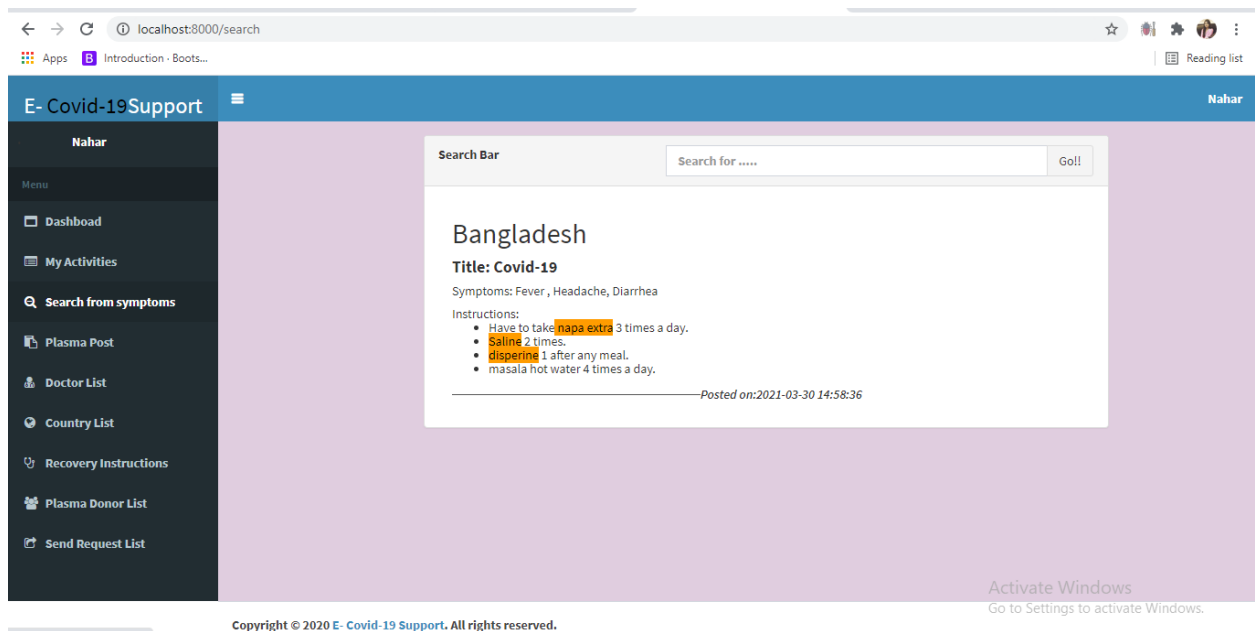
Country	Title	Syntomps of covid-19	Recovery Instruction	Recommand by	Status	Action
Bangladesh	Covid-19	Fever , Headache, Diarrhea	<ul style="list-style-type: none">Have to take napa extra 3 times a day.Saline 2 times.disperine 1 after any meal.masala hot water 4 times a day.	Dr. Liza	Active	Details
Canada	Covid-19 Fever	High fever for more than 7 days, smell less feeling, skin rashes	<ol style="list-style-type: none">napa Extra after 8 hours until fever recovered.Altertrol for rashes.Hot water 4 times a day.	Dr. Tom	Active	Details

At the bottom of the page, there is a copyright notice: 'Copyright © 2020 E- Covid-19 Support. All rights reserved.' and an 'Activate Windows' watermark.

9. Registered user can comment, like or dislike on recovery instruction post.



10. Patient can search from his symptoms.



11. Patient can create plasma post.

The screenshot shows a web browser at localhost:8000/plasmaposts. The application has a dark sidebar menu with options: Dashboard, My Activities, Search from symptoms, Plasma Post, Doctor List, Country List, Recovery Instructions, Plasma Donor List, and Send Request List. The main content area is titled "Plasma Post List" and features a green "+ Add Plasma Post" button. Below the button is a table with two rows of plasma posts.

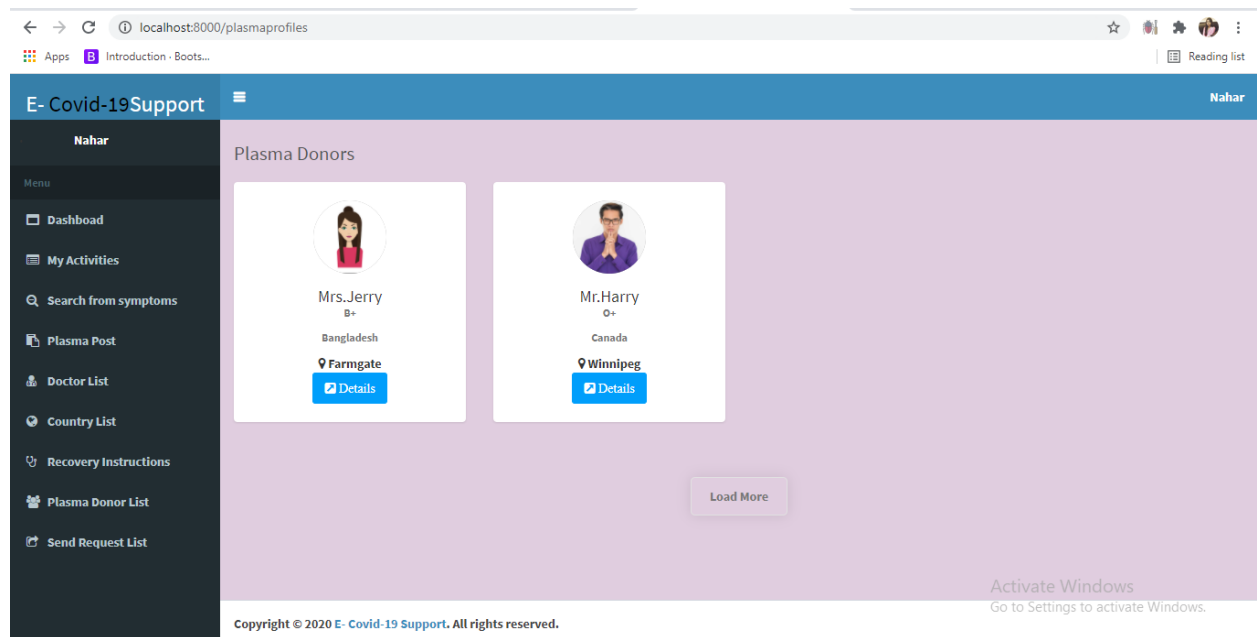
Country	Title	Contact & Address	Blood Group	Recommend by	Action
USA	Plasma O+	Need plasma urgently for my uncle. Contact: +1 2345 642 Location: satint lorence hospital, 2nd floor.	O+	kamrun	Details
Bangladesh	Urgently Need Plasma	Contact: 0198642735 Address: Holy Family Hospital, 3rd floor.	B+	Nahar	Details

At the bottom of the page, there is a copyright notice: "Copyright © 2020 E- Covid-19 Support. All rights reserved." and an "Activate Windows" watermark.

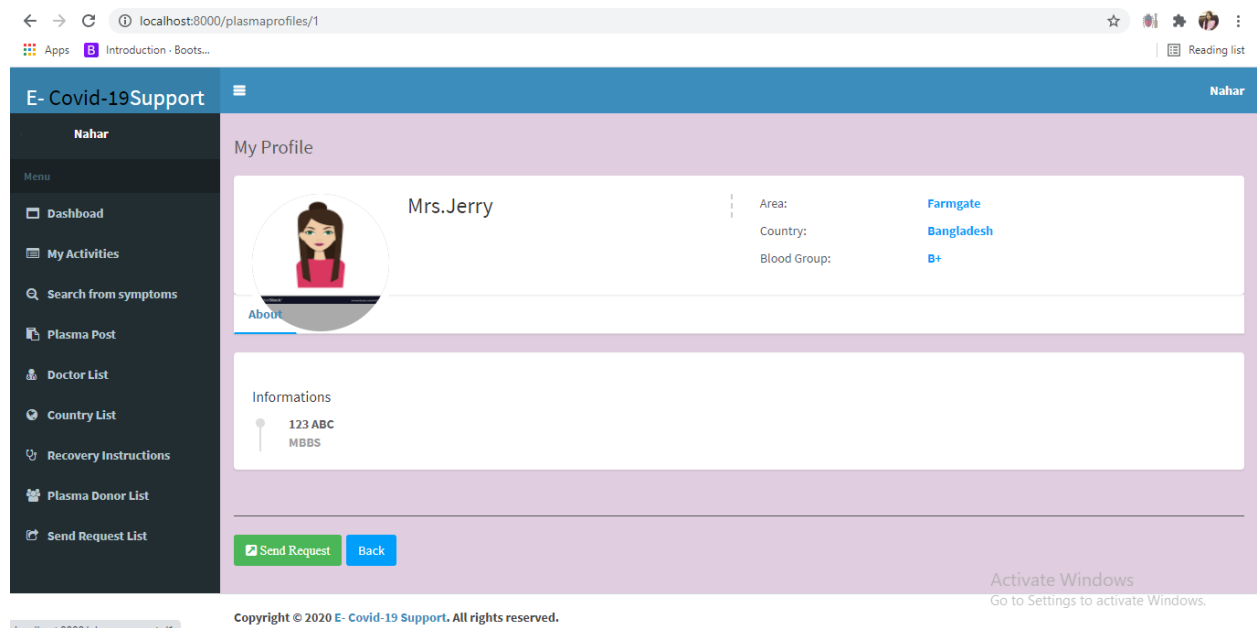
12. Other patient and plasma donor can comment on plasma post.

The screenshot shows the same web application, but now displaying a single plasma post. The post details are: Title: Plasma O+, Blood Group: O+, Contact & Address: Need plasma urgently for my uncle. Contact: +1 2345 642 Location: satint lorence hospital, 2nd floor., Posted By: kamrun, Country: USA, and Posted at: 14 hours ago. Below the post details is a "Comment" section with a text input field and a green "Comment" button. Underneath the input field, there is a "Comments" section showing a comment from "Harry:" with the text "I will come to give you plasma." and a timestamp of "Commented at: 14 hours ago".

13. They can see plasma donor list.



14. They can send request to plasma donor if he needs plasma urgently.



15. After send request plasma donor can see the list who has send him request for plasma if he want to give he will accept the request.

The screenshot shows a web browser at localhost:8000/receive. The application header is 'E- Covid-19Support' with a user profile 'Jerry'. A sidebar menu on the left includes options like Dashboard, Plasma Post, Doctor List, Country List, Recovery Instructions, Plasma Donor List, and Plasma Request List (which is selected). The main content area is titled 'Get Request List' and contains a table with the following data:

Blood Group	Location	Contact	Message	Send By	Request To	Status	Action
B+	Farmgate	01798765432	I want plasma.	Nahar	Mrs. Jerry	Pending	Accept

At the bottom of the page, there is a copyright notice: 'Copyright © 2020 E- Covid-19 Support. All rights reserved.' and an 'Activate Windows' watermark.

16. After send request he can see get request list he has send to plasma donors. If plasma donor accept patient can see his all details.

The screenshot shows a web browser at localhost:8000/plasmarequests. The application header is 'E- Covid-19Support' with a user profile 'Nahar'. The sidebar menu on the left includes options like Dashboard, My Activities, Search from symptoms, Plasma Post, Doctor List, Country List, Recovery Instructions, Plasma Donor List, and Send Request List (which is selected). The main content area is titled 'Send Request List' and contains a table with the following data:

Blood Group	Location	Contact	Message	Request To	Status	Action
B+	Farmgate	01798765432	I want plasma.	Mrs. Jerry	Accepted	Details
O+	Winnipeg	01798765432	I want plasma	Mr. Harry	Pending	Waiting for accept

At the bottom of the page, there is a copyright notice: 'Copyright © 2020 E- Covid-19 Support. All rights reserved.' and an 'Activate Windows' watermark.

CHAPTER-07

Project Summary

7.1 GitHub Link: github.com/Kamrun-Nahar-Liza/E-Covid-19-Support

7.2 Limitations: Cannot do private conversation with each other.

7.3 Obstacles and Achievements:

Everyday technologies are updated so the Laravel framework is updated every day. So, I features there are some technologic can be eliminating and the project can be implemented with new technology. There are too more errors while I am developed this system.

7.4 Future Work:

Every Project has some future scope so that in features the application adds some new feature:

1. Can do private chat with each other.
2. Patients can have appointment with doctors.

APPENDICES-A

```
<?php

namespace App\Http\Controllers;

use Auth;
use App\Post;
use Validator;
use App\Comment;
use App\Country;
use App\Like;
use App\Dislike;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class PostController extends Controller
{

    public function index()
    {
        $data=[];
        $data['posts'] = Post::with('country','user')->select('id','title', 'symptoms'
        ,'content','user_id','country_id','status')->get();

        return view('post.index', $data);
    }

    public function create()
    {
        $data=[];
        $data['countries']=Country::select('id','name','code','created_at')->get();
        return view('post.create',$data);
    }

    public function store(Request $request)
    {
        //validation

        $rules= [
            'title' => 'required',
            'symptoms' => 'required',
```

```

        'content' => 'required',
        'status' => 'required',
        'country_id' => 'required'

];

$validator = Validator::make($request->all(), $rules);

if ($validator->fails()) {
    return redirect()->back()->withErrors($validator)->withInput();
}

//database insert

Post::create([

    'title' => $request->input('title'),
    'symptoms' => $request->input('symptoms'),
    'content' => $request->input('content'),
    'status' => $request->input('status'),
    'country_id' => $request->input('country_id'),
    'user_id' => auth()->user()->id
]);

//redirect

session()->flash('message','Recovery Post Added Successfully');
session()->flash('type','success');
return redirect()->route('posts.create');
}

public function show($id)
{
    $data=[];
    $data['post'] = Post::with('country','user')->select('id','title', 'symptoms',
    'content','user_id','country_id','status','created_at')->find($id);

    // $post = Post::with('country','user')->where('id', '=', $post_id)->get();
    // $likePost= Post::find($post_id);
    // $likeCtr = Like::where(['post_id' => $likePost->id])->count();
    // $dislikeCtr = DisLike::where(['post_id' => $likePost->id])->count();

    $likePost= Post::find($id);
    $likeCtr = Like::where(['post_id' => $likePost->id])->count();

```



```

$dislikeCtr = DisLike::where(['post_id' => $likePost->id])->count();

$comments = DB::table('users')
    ->join('comments', 'users.id', '=', 'comments.user_id')
    ->join('posts', 'comments.post_id', '=', 'posts.id')
    ->select('users.name', 'comments.*')
    ->where(['posts.id' => $id])
    ->get();

    return view('post.show', ['comments' => $comments, 'likePost' => 'likePost', 'likeCtr' =>
$likeCtr, 'dislikeCtr' => $dislikeCtr], $data);
}

public function edit($id)
{
    $data=[];
    $data['post'] = Post::with('country', 'user')->select('id', 'title', 'symptoms',
'content', 'user_id', 'country_id', 'status', 'created_at')->find($id);
    $data['countries']=Country::select('id', 'name', 'code', 'created_at')->get();
    return view('post.edit', $data);
}

public function update(Request $request, $id)
{
    //validation

    $rules= [
        'title' => 'required',
        'symptoms' => 'required',
        'content' => 'required',
        'status' => 'required',
        'country_id' => 'required'

    ];

    $validator = Validator::make($request->all(), $rules);

    if ($validator->fails()) {
        return redirect()->back()->withErrors($validator)->withInput();
    }
}

```

```

//database update

$post = Post::find($id);

$post->update([

    'title' => $request->input('title'),
    'symptoms' => $request->input('symptoms'),
    'content' => $request->input('content'),
    'country_id' => $request->input('country_id'),
    'status' => $request->input('status')
]);

//redirect

session()->flash('message','post has been updated');
session()->flash('type','success');
return redirect()->back();
}

public function destroy($id)
{
    $post = Post::find($id);
    $post->delete();

    //redirect

    session()->flash('message','Post deleted');
    session()->flash('type','danger');
    return redirect()->route('posts.index');
}

//search option code

public function searchboard()
{
    $data=[];
    return view('search.searchboard' , $data);
}

public function search(Request $request){

    $user_id = Auth::user()->id;

    $keyword = $request->input('search');

```

```

    $posts = Post::where('symptoms', 'LIKE', '%'.$keyword.'%')->get();
    return view('search.searchposts', ['posts' => $posts]);
}

//comment code

public function comment(Request $request, $post_id){

    $this->validate($request, [
        'comment' => 'required'
    ]);

    $comment = new Comment;
    $comment->user_id = Auth::user()->id;
    $comment->post_id = $post_id;
    $comment->comment = $request->input('comment');
    $comment->save();

    return redirect()->back()->with('response', 'Comment added successfully');
}

public function delete($id)
{
    $comment = Comment::find($id);
    $comment->delete();

    //redirect
    return redirect()->back()->with('response', 'Comment added successfully');
}

//Like

public function like($id){
    $loggedin_user = Auth::user()->id;
    $like_user = Like::where(['user_id' => $loggedin_user, 'post_id' => $id])->first();
    if(empty($like_user->user_id)){
        $user_id = Auth::user()->id;
        $email = Auth::user()->email;
        $post_id = $id;
        $like = new Like;
        $like->user_id= $user_id;
        $like->email= $email;
        $like->post_id= $post_id;
        $like->save();
        return redirect("/posts/{ $id}");
    }
}

```

```

    }

    else{
        return redirect("/posts/{ $id }");
    }
}

//dislike

public function dislike($id){
    $loggedin_user = Auth::user()->id;
    $like_user = Dislike::where(['user_id' => $loggedin_user , 'post_id' => $id])->first();
    if(empty($like_user->user_id)){
        $user_id = Auth::user()->id;
        $email = Auth::user()->email;
        $post_id = $id;
        $like = new Dislike;
        $like->user_id= $user_id;
        $like->email= $email;
        $like->post_id= $post_id;
        $like->save();
        return redirect("/posts/{ $id }");
    }

    else{
        return redirect("/posts/{ $id }");
    }
}

}

```

APPENDICES-B

```
<?php

namespace App\Http\Controllers;

use DB;
use App\Post;
use App\User;
use App\Covid;
use App\Country;
use App\Profile;
use App\PlasmaPost;
use App\PlasmaProfile;
use Illuminate\Http\Request;

class HomeController extends Controller
{
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    // public function __construct()
    // {
    //     $this->middleware('auth');
    // }

    /**
     * Show the application dashboard.
     *
     * @return \Illuminate\Contracts\Support\Renderable
     */

    public function homepage()
    {

        $data = DB::table('covids')
            ->select('*')
            ->orderBy('id','desc')
            ->limit(1)
            ->get();

        // (select infect from covids ORDER BY id DESC LIMIT 1);
    }
}
```

```

$totalinfect = DB::table('covids')
    ->sum('infect');
    // ->get();

$totaldeath = DB::table('covids')
    ->sum('death');

$totalcure = DB::table('covids')
    ->sum('cure');

$totaltest = DB::table('covids')
    ->sum('test');

// dd($totalinfect);

return view ('homepage', compact('data', 'totalinfect','totaldeath','totalcure','totaltest'));

}

public function index()
{
    return view('home');
}

public function doctorindex()
{
    $data = DB::table('covids')
        ->select('*')
        ->orderBy('id','desc')
        ->limit(1)
        ->get();

    // (select infect from covids ORDER BY id DESC LIMIT 1);

    $totalinfect = DB::table('covids')
        ->sum('infect');
        // ->get();

    $totaldeath = DB::table('covids')
        ->sum('death');

    $totalcure = DB::table('covids')
        ->sum('cure');

```

```

$totaltest = DB::table('covids')
    ->sum('test');

// dd($totalinfect);

return view ('doctor', compact('data', 'totalinfect','totaldeath','totalcure','totaltest'));
}

public function patientindex()
{
    $data = DB::table('covids')
        ->select('*')
        ->orderBy('id','desc')
        ->limit(1)
        ->get();

    // (select infect from covids ORDER BY id DESC LIMIT 1);

    $totalinfect = DB::table('covids')
        ->sum('infect');
    // ->get();

    $totaldeath = DB::table('covids')
        ->sum('death');

    $totalcure = DB::table('covids')
        ->sum('cure');

    $totaltest = DB::table('covids')
        ->sum('test');

    return view('patient', compact('data', 'totalinfect','totaldeath','totalcure','totaltest'));
}

public function plasmadonorindex()
{
    $data = DB::table('covids')
        ->select('*')
        ->orderBy('id','desc')
        ->limit(1)
        ->get();

    // (select infect from covids ORDER BY id DESC LIMIT 1);

    $totalinfect = DB::table('covids')

```

```

        ->sum('infect');
        // ->get();

$totaldeath = DB::table('covids')
    ->sum('death');

$totalcure = DB::table('covids')
    ->sum('cure');

$totaltest = DB::table('covids')
    ->sum('test');

return view('plasmadonor', compact('data', 'totalinfect', 'totaldeath', 'totalcure', 'totaltest'));
}

public function adminindex()
{
    $data = DB::table('covids')
        ->select('*')
        ->orderBy('id', 'desc')
        ->limit(1)
        ->get();

    // (select infect from covids ORDER BY id DESC LIMIT 1);

    $totalinfect = DB::table('covids')
        ->sum('infect');
        // ->get();

    $totaldeath = DB::table('covids')
        ->sum('death');

    $totalcure = DB::table('covids')
        ->sum('cure');

    $totaltest = DB::table('covids')
        ->sum('test');

    return view('admin', compact('data', 'totalinfect', 'totaldeath', 'totalcure', 'totaltest'));
}

public function service()
{
    return view('Homepages.service');
}

```



```

}

public function doctorlist()
{
    $data=[];
    $data['profiles'] = Profile::all();
    return view('Homepages.doctorlist',$data);
}

public function donorlist()
{
    $data=[];
    $data['plasmaprofiles'] = PlasmaProfile::all();
    return view('Homepages.donorlist',$data);
}

public function doctor_activity()
{
    $data['posts'] = Post::all();
    $data['countries'] = Country::all();
    return view('activity.doctor_activity', $data);
}

public function patient_activity()
{
    $data['plasmaposts'] = PlasmaPost::all();
    $data['countries'] = Country::all();
    return view('activity.patient_activity',$data);
}

```

// Doctor Verification

```

public function Accept(Request $request)
{
    $id = $request->id;
    $user = User::find($id);
    $user->is_doctor = '1';
    $user->save();
}

public function Pending(Request $request)
{
    $id = $request->id;

```

```

    $user = User::find($id);
    $user->is_doctor = '0';
    $user->save();
}

//verify list

    public function verify(){

        $newdata = DB::table('users')
        ->select('*')
        ->where('role', '=', 'doctor')

        ->get();

        return view('activity.verify_doctor', compact('newdata'));

    }

//doctor_delete
    public function doctor_delete($id){
        $user = User::find($id);
        $user->delete();

        //redirect
        return view('activity.verify_doctor');
    }

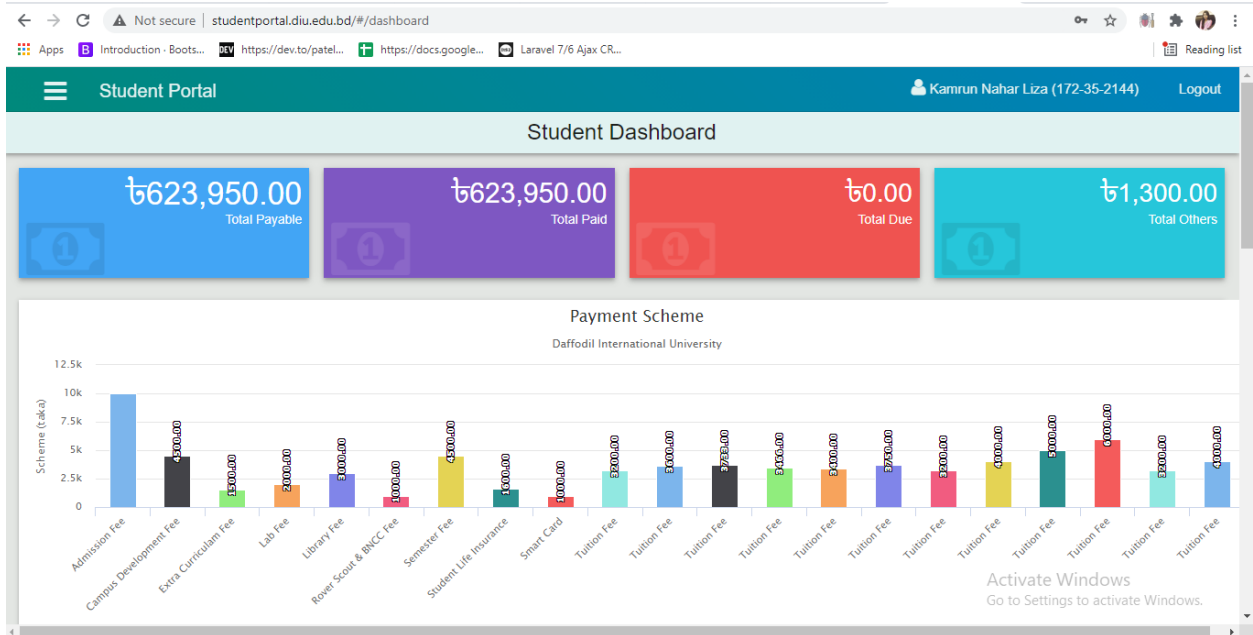
}

```

REFERENCES

1. <https://mogadalai.wordpress.com/2007/05/28/howto-write-a-project-proposal/> (Accessed on: 01-09-2020)
2. <https://stackoverflow.com/questions/29582625/send-a-button-value-with-jquery-ajax> (Accessed on: 31-03-2021)
3. <https://themeforest.net/item/prolexe-health-and-medical-html-template> (Accessed on: 31-03-2021)
4. <https://www.w3schools.com/> (Accessed on: 31-03-2021)
5. <https://laravel.com/docs/5.8> (Accessed on: 31-03-2021)
6. <https://stackoverflow.com/questions/3320283/manage-user-and-roles>. (Accessed on: 31-03-2021)
7. <https://laracasts.com/discuss/channels/laravel/current-date-in-laravel> (Accessed on: 31-03-2021)
8. <https://summernote.org/getting-started/> (Accessed on: 31-03-2021)
9. <https://guides.lib.berkeley.edu/how-to-write-good-documentation> (Accessed on: 31-03-2021)
10. <https://www.sitepoint.com/writing-software-documentation/> (Accessed on: 31-03-2021)

Clearance:



Student Portal | Kamrun Nahar Liza (172-35-2144) | Logout

Registration/Exam Clearance

Semester	Registration	Mid Term Exam	Final Exam/Assessment
Summer, 2017	✓	✓	✓ (Final Exam)
Fall, 2017	✓	✓	✓ (Final Exam)
Spring, 2018	✓	✓	✓ (Final Exam)
Summer, 2018	✓	✓	✓ (Final Exam)
Fall, 2018	✓	✓	✓ (Final Exam)
Spring, 2019	✓	✓	✓ (Final Exam)
Summer, 2019	✓	✓	✓ (Final Exam)
Fall, 2019	✓	✓	✓ (Final Exam)
Spring, 2020	✓	✓	✗ (Final Exam)
Summer, 2020	✓	✓	✓ (Final Exam)
Fall, 2020	✓	✗	✓ (Final Exam)

Activate Windows
Go to Settings to activate Windows.